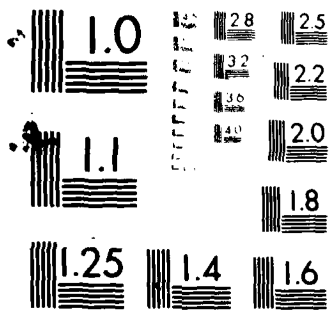
AD-A109 861   NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIRCRAFT --ETC F/G 1/3
A SHORT TAKEOFF PERFORMANCE COMPUTER PROGRAM.(U)
NOV 81 D B KOBUS
UNCLASSIFIED NADC-81259-60   NL

1 of 2



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

REPORT NO. NADC-81259-60

(12)

LEVEL II

AD A109861

A SHORT TAKEOFF PERFORMANCE COMPUTER PROGRAM

David Bruce Kobus
Aircraft and Crew Systems Technology Directorate
NAVAL AIR DEVELOPMENT CENTER
Warminster, Pennsylvania

DTIC
ELECTE
JAN 21 1982
S E

25 NOVEMBER 1981

FINAL REPORT
AIRTASK NO. ZR020302
Work Unit No. GC172

Approved for Public Release; Distribution Unlimited

Prepared for
NAVAL AIR SYSTEMS COMMAND
Department of the Navy
Washington, DC 20361

01 20 82025

NOTICES

REPORT NUMBERING SYSTEM - The numbering of technical project reports issued by the Naval Air Development Center is arranged for specific identification purposes. Each number consists of the Center acronym, the calendar year in which the number was assigned, the sequence number of the report within the specific calendar year, and the official 2-digit correspondence code of the Command Office or the Functional Directorate responsible for the report. For example: Report No. NADC-78015-20 indicates the fifteeth Center report for the year 1978, and prepared by the Systems Directorate. The numerical codes are as follows:

CODE	OFFICE OR DIRECTORATE
00	Commander, Naval Air Development Center
01	Technical Director, Naval Air Development Center
02	Comptroller
10	Directorate Command Projects
20	Systems Directorate
30	Sensors & Avionics Technology Directorate
40	Communication & Navigation Technology Directorate
50	Software Computer Directorate
60	Aircraft & Crew Systems Technology Directorate
70	Planning Assessment Resources
80	Engineering Support Group

PRODUCT ENDORSEMENT - The discussion or instructions concerning commercial products herein do not constitute an endorsement by the Government nor do they convey or imply the license or right to use such products.

APPROVED BY: J. R. Woods DATE: 11/30/81

J. R. WOODS
CDR USN

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: NADC-81259-60	2. GOVT ACCESSION NO.: AD-A109861
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): A Short Takeoff Performance Computer Program	5. TYPE OF REPORT & PERIOD COVERED: Final Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): DAVID BRUCE KOBUS	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Naval Air Development Center, Aircraft & Crew Systems Technology Directorate, Warminster, PA 18974	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: AIRTASK NO. ZR020302, Work Unit No. GC172
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Air Systems Command, Department of the Navy, Washington, DC 20361	12. REPORT DATE: 25 November 1981
	13. NUMBER OF PAGES: 120
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

V/STOL
Takeoff
STO

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A short takeoff performance computer program has been developed by NAVAIRDEVCEN to be used for performance estimation and conceptual aircraft design.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SUMMARY

A computer program has been developed which is capable of analyzing the short takeoff of typical V/STOL aircraft configurations. This program can be used as a means for performance estimation as well as for assisting in aircraft conceptual design.

This program was written in FORTRAN for use on a CDC 6600 and uses five supporting subroutines. This report describes the analytical development and logic development for the program. In addition it includes a user description and complete listing of the program

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a_x, a_z	Longitudinal and vertical acceleration (m/sec^2, or ft/sec^2)
a/g	Normalized longitudinal acceleration
C_D	Drag coefficient
C_L	Lift coefficient
C_M	Pitching moment coefficient
D	Drag (N or lb f)
F_G	Gross thrust (N or lb f)
F_{G_F}, F_{G_A}	Fore and aft engine gross thrust (N or lb f)
F_{GH}, F_{G_V}	Horizontal and vertical gross thrust (N or lb f)
g	Acceleration due to gravity (9.8 m/sec^2 or 32.2 ft/sec^2)
$I_{yy_{cg}}$	Lateral moment of inertia (kg-m^2 or slug-ft^2)
L	Lift (N or lb f)
My	Lateral pitching moment (N-m or ft-lb f)
m	Aircraft mass (kg or lb)
R_F, R_A	Fore and aft gear wheel reaction force (N or lb f)
RD_F, RDA	Fore and aft engine ram drag (N or lb f)
S_x, S_z	Longitudinal and vertical distance (m or ft)
T	Time (sec)
V_x, V_z	Longitudinal and vertical velocity (m/sec or ft/sec)
WOD	Wind over deck
W_F	Fuel flow (kg/hr or lb/hr)
W	Aircraft weight (m*g)
α	Angle of attack (deg)
α_{MG}	Angular acceleration about main gear wheel (deg/sec^2)
γ	Flight path angle (deg)

LIST OF SYMBOLS

Δ	Increment
δ_e	Control surface deflection angle (deg)
θ_j	Jet nozzle inclination angle (deg)
θ_p	Pitch Angle (deg)
μ	Coefficient of friction
Σ	Sum

INTRODUCTION

A methodology for predicting the short takeoff (STO) performance of Vertical or Short Takeoff and Landing (V/STOL) aircraft is essential for the preliminary design of these vehicles as the propulsion system may be sized by STO requirements. Since a wide range of VSTOL configurations may be of interest to the Navy, and each would have its own unique set of thrust and drag variations during the takeoff run, a generalized computer program offers the most feasible method of analyzing STO performance.

Several STO programs currently available in the technical community were examined for applicability. It was found that each program was developed for a specific configuration and did not provide the flexibility necessary to examine arbitrary V/STOL aircraft concepts. Since the existing programs could not be easily modified, it was necessary to develop a new program.

The resulting program can be utilized for performance estimation and for aircraft conceptual design. In the performance estimation mode, inputs are aircraft aerodynamics, propulsion data, takeoff gross weight (TOGW), and STO technique while outputs are time, distance, velocity, longitudinal acceleration, lift, drag, and horizontal and vertical gross thrust. These data can be calculated at the condition of liftoff, at some specified rate of climb (R/C), or at some specified sink distance. In the conceptual design mode, inputs are given design parameters such as takeoff distance, rate of climb at liftoff, and longitudinal acceleration (with or without one engine inoperative (OEI)), and outputs such as engine size and optimum STO technique can be determined.

DISCUSSION

Theoretical Considerations

This program utilizes an open form, time history, numerical integration. The forces included in this treatment are lift, drag, gross thrust, and friction. A force diagram is presented in Figure 1. The horizontal forces are summed and divided by the aircraft mass to yield a horizontal acceleration. This acceleration in turn is equated to the following kinematic formula:

$$a_x = \frac{\Delta V_x}{\Delta T} = \frac{1}{2}\,\frac{\Delta(V_x{}^2)}{\Delta S_x}$$

Using time increments, a change in velocity and distance can be determined. More specifically, the basic equations are:

$$DRAG = C_D*1/2\rho V^2 S + D_{ENGINE\ RELATED} + D_{STORE} + D_{OEI} + D_{INDUCED}$$

where: $D_{ENGINE\ RELATED}$ = ram drag, spillage drag, boattail drag, etc.

$$LIFT = C_L*1/2\rho V^2 S + L_{INDUCED}$$

Summing vertical gross thrust components:

$$F_{G_V} = F_{G_{F_V}} + F_{G_{A_V}}$$

$$F_{G_V} = F_{G_1}*SIN\ \beta_1 + F_{G_2}*SIN\ \beta_2$$

where: $F_{G1} = F_{G_F}$ - fore engine gross thrust loss

$F_{G2} = F_{GA}$ - aft engine gross thrust loss

$\beta_1 = \theta_{jF} + \theta_p$ + engine datum angle

$\beta_2 = \theta_{jA} + \theta_p$ + engine datum angle

Summing horizontal gross thrust components:

$$F_{G_H} = F_{G_{F_H}} + F_{G_{A_H}}$$

$$F_{G_H} = F_{G1}*COS\ \beta_1 + F_{G_2}*COS\ \beta_2$$

Balancing the vertical forces:

$$BALANCE = W - Lift - F_{G_V}$$

The change in velocity over a time increment ΔT is:

$$\Delta V_x = \Delta T*g*(F_{GH} - DRAG - \mu *BALANCE)/W$$

where: $\Delta V_x = V_{x2} - V_{x1}$

Finally, the distance gained over ΔT is:

$$\Delta S_x = 1/2*\Delta T*((V_{x_2}-WOD)^2 - (V_{x_1}-WOD)^2)/\Delta V_x$$

By using this technique the forces which vary as a function of time and angle can be accurately represented and, in addition, pilot technique can also be closely approximated. Also by using small time increments any procedural changes can be fitted through the input.

Increased program accuracy can be obtained by accounting for and by correcting a pitching moment inbalance. This is done through the following formula:

$$\Sigma\ My = I_{yy_{cg}}\ \alpha_{MG}$$

By making use of the supplied aircraft moment of inertia and a power-on pitching moment, an angular acceleration about the main gear can be taken into account. The moment about the vehicle center of gravity is:

$$MOMENT = C_M * 1/2\rho V^2*MAC$$

The fore gear wheel reaction force is therefore:

$$R_F = (BALANCE*\beta_3-MOMENT)/(LRF + \beta_3 + \cos\ \theta_p)$$

where: β_3 = LRA*cos θ_p+ LRA*μ*sin θ_p+ LR*μ*cos θ_p - LR*sin θ_p

See Table I for the definitions of LR, LRA, LRF and MAC.

If the supplied aerodynamic data is a function of control surface deflection, the deflection required to nullify any moments can be calculated. This is accomplished by the following algorithm:

initialize α, θ_j and δe
$C_L = f(\alpha, \delta e, \theta_j)$
$C_D = f(C_L, \delta e, \theta_j)$
$C_M = f(C_L, \delta e, \theta_j)$

$$DRAG = f(C_D) + D_{ENGINE\ RELATED} + D_{STORE} + D_{OEI} + D_{INDUCED}$$

$$LIFT = f(C_L) + L_{INDUCED}$$

$$MOMENT = f(C_M)$$

$$F_{GV} = f(F_G, \theta_j, \theta_p)$$

$$BALANCE = W - Lift - F_{GV}$$

$$R_F = f\ (BALANCE, MOMENT)$$

iterate δe until $R_F = 0$

Rates of climb or sink can be calculated if the horizontal component of velocity is considered much larger than the vertical component ($V_z < .02\ V_x$). Here the vertical acceleration is: $a_z = \dfrac{\Delta V_z}{\Delta T}$

and the vertical velocity is: $V_z = \dfrac{\Delta S_z}{\Delta T}$

Hence, climb or sink accelerations, velocities, and distances can be determined. For rate of climb velocity over the ΔT increment:

$$a_z = (-BALANCE)*g/w$$

$$\bar{a}_z = 1/2*(a_{z2} + a_{z1})$$

To determine the incremental ROC:

$$\Delta ROC = \bar{a}_z*\Delta T$$

For sink off the bow distance over the ΔT increment:

$$a_z = BALANCE*g/w$$

$$\bar{a}_z = 1/2*(a_{z2} + a_{z1})$$

$$\Delta V_z = \bar{a}_z*\ \Delta T$$

$$V_{z2} = V_{z1} + \Delta V_z$$

$$\overline{V}_z = 1/2 *(V_{z2} + V_{z1})$$

The incremental sink distance is:

$$\Delta SINK = \overline{V}_z *\Delta T$$

If an instantaneous ROC at liftoff is needed, the following formula is used:

$$a_x = (F_{G_H} - DRAG - \mu*BALANCE)/W$$

Assuming a small flight path angle at liftoff:

$$ROC_{instantaneous} = V \sin \gamma \approx V\gamma$$

Where $V\gamma = V *((LIFT + F_{GV} - W)/F_{G_H})$

The main program logic, which illustrates how all the above equations are implemented, is presented in Appendix D.

Program Description

In addition to the main program, which performs the computations on the dynamic equations, there are five supporting subroutines, as described below.

TAKE 5 - In the formulation of the STO computer program a four degree-of-freedom interpolation routine was developed from an earlier three degree-of-freedom routine (reference (a)). This was done to accomodate aerodynamic inputs which are a function of four variables.

This subroutine has two modes of operation. In the first mode, table data representing aerodynamic and propulsion characteristics are input and stored. Each table is assigned a predetermined reference number. In the second mode, table data is interpolated and extrapolated by the function SPLNQ1 for use in the dynamic calculations. TAKE5 subroutine logic is presented in Appendix D.

SPLNQ1 - This function is used to interpolate or extrapolate tabular data. The interpolation is calculated using a local curve fit scheme described in reference (b). Linear extrapolations are made using each end point slope of the local curve fit.

UPDATE - This subroutine is used to reshape data arrays into a form usuable by the function SPLNQ1.

SNEST - This is an iteration subroutine used to solve one degree-of-freedom problems. A slope intercept convergence technique is employed.

ATMOS - This is a 1962 standard atmosphere table which returns properties of density, pressure, temperature, and sound velocity for an input geometric altitude.

The data required for the program consists of a series of single-value, fixed inputs and multiple-valued tabular inputs. The form of the computer data deck necessary to make a run is presented in Figure 2.

The tabular data includes:

- Gross thrust as a function of Mach number and engine speedup time.

- Fuel flow as a function of Mach number and engine speedup time.

- Drag coefficient as a function of lift coefficient, control surface deflection, jet nozzle inclination, and velocity.

- Lift coefficient as a function of angle of attack, control surface deflection, jet nozzle inclination, and velocity.

- Induced lift increment as a function of jet inclination angle

- Induced drag increment as a function of jet inclination angle

- Pitching moment coefficient as a function of lift coefficient, control surface deflection, jet nozzle inclination, and velocity.

- Aircraft moment of inertia as a function of aircraft mass.

- Distance from aircraft center of gravity to main gear wheel as a function of aircraft mass.

- Gross thrust loss relative to zero nozzle deflection gross thrust as a function of jet nozzle angle.

- Throttle dependent, ram, and external store drag as a function of Mach number.

These tabular inputs are graphically illustrated in Appendix A.

The fixed inputs consist of aircraft size and mass data, ambient conditions, initial thrust vector angles and control surface deflections, rotation rates for the nozzles and specification for events that may occur during the takeoff run (e.g. loss of an engine). Table I contains a fixed input variable list.

Program Options

Available to the user are two main options which concern the type of STO vehicle that can be approximated. First, is a Harrier type vehicle which maintains moment balance by a reaction control system, rotates its jet nozzles, and pitches up at a given rate and at a predetermined airspeed. Second, is a vehicle which maintains moment balance by a control surface deflection and has a fixed jet nozzle angle.

For vehicles using control surfaces to maintain moment balance on the ground, the program will search for the start of auto rotation (stick free pitch up) and either give information on the rotation of the aircraft about the aft wheel or attempt to correct this rotation by the deflection of a control surface. Pertinent tabular inputs for this vehicle type include:

$$C_M = f(C_L, \delta_e, \theta_j, V_x)$$

$$I_{yyCG} = f(m)$$

Distance from CG to aft wheel = f(m)

Fixed inputs concerned with this type of vehicle are LR, LRA, LRF, DELTAE, VSTARR, MAC, SEGMENT, IYYCG, XCG, YCG, DELMAX and ALPHMIN. Examination of the above data, by considering the descriptions made in Table I and Appendix A, will clearly illustrate the options available for this type of vehicle.

If a Harrier type vehicle is to be considered, the previously mentioned input can be ignored.

By viewing the form of the remaining tabular and fixed input, in Table I and Appendix A, it is seen that a wide variety of pilot techniques can be approximated. This is because any continuous or discontinuous tabular aerodynamic or propulsion function can be fitted. In addition, a large variety of fixed input switches are available for duplication of aircraft rotation and thrust deflection at any prescribed airspeed.

The following comments offer some additional options that can be implemented for a Harrier type vehicle.

1. If a series of data cases are to be run, put the additional aircraft weights or velocities in ascending order. Set RUN not equal to zero for interpolated weight or set RUN and PSINT not equal to zero for interpolated pitch speed.

2. Apply variables RS, ALPHARR, ALPHAAR, ALPHA, PHI2, PHI2AR, and PS for desired rotation method. For the fore engine nozzle variation apply RSF, PHI1, PHI1AR, PHI1RR accordingly.

3. Set NOEDRAG not equal to zero and NOSTORE not equal to zero if engine related drags and store drags exist in reference table 33 (nos. 1 and 2) (see Appendix A). Set NODOEI not equal to zero if engine out drag exists in reference table 33 (no. 3).

4. For OEI at brake release, read in NEF = NEA = 0.5 or any appropriate scale down, or put in single engine tabular data inputs and add failed engine drag in store drag (reference table 33 (no. 2)). For OEI at liftoff, set EFAILA not equal to zero and OEILOSF to an appropriate front engine scale down value and add failed engine drag as additional drag (reference table 33 (no. 3)). Set OEITAB not equal to zero for OEI at liftoff when thrust, fuel flow, and drags (engine related and external store) for one engine out are supplied in tabular form (reference table 10, 11, and 33 (nos. 3 and 4)). Put in RAM for a constant ram drag scale down during OEI.

5. For dynamic R/C constraint, put in ROC limit. Set PHONY not equal to zero when instantaneous R/C is desired (fore and aft nozzle angles and aircraft pitch angle are set to maximum values for instantaneous R/C calculation). Set SINKD to bow height for sink calculations. (If PS, RS, and RSF have not been reached at SINKD, these variables will be set to the airspeed at SINKD). NOTE: rate of climb and sink estimates assume that the horizontal velocity is much greater than the vertical velocity.

ACKNOWLEDGEMENTS

A mention of appreciation is given by the author to Michael Caddy who developed the supporting subroutines and to John Cyrus for his technical concepts.

REFERENCES

(a) Caddy, M. J., "TREAD/TLOOK - Multipurpose Computer Routine for Interpolation and Extrapolation of Tabular Data," NAVAIRDEVCEN Report No. NADC-76366-30, 11 January 1977

(b) Akima, Hiroshi, "Interpolation and Smooth Curve Fitting Based on Local Procedures," Institute for Telecommunication Sciences, 1 March 1972

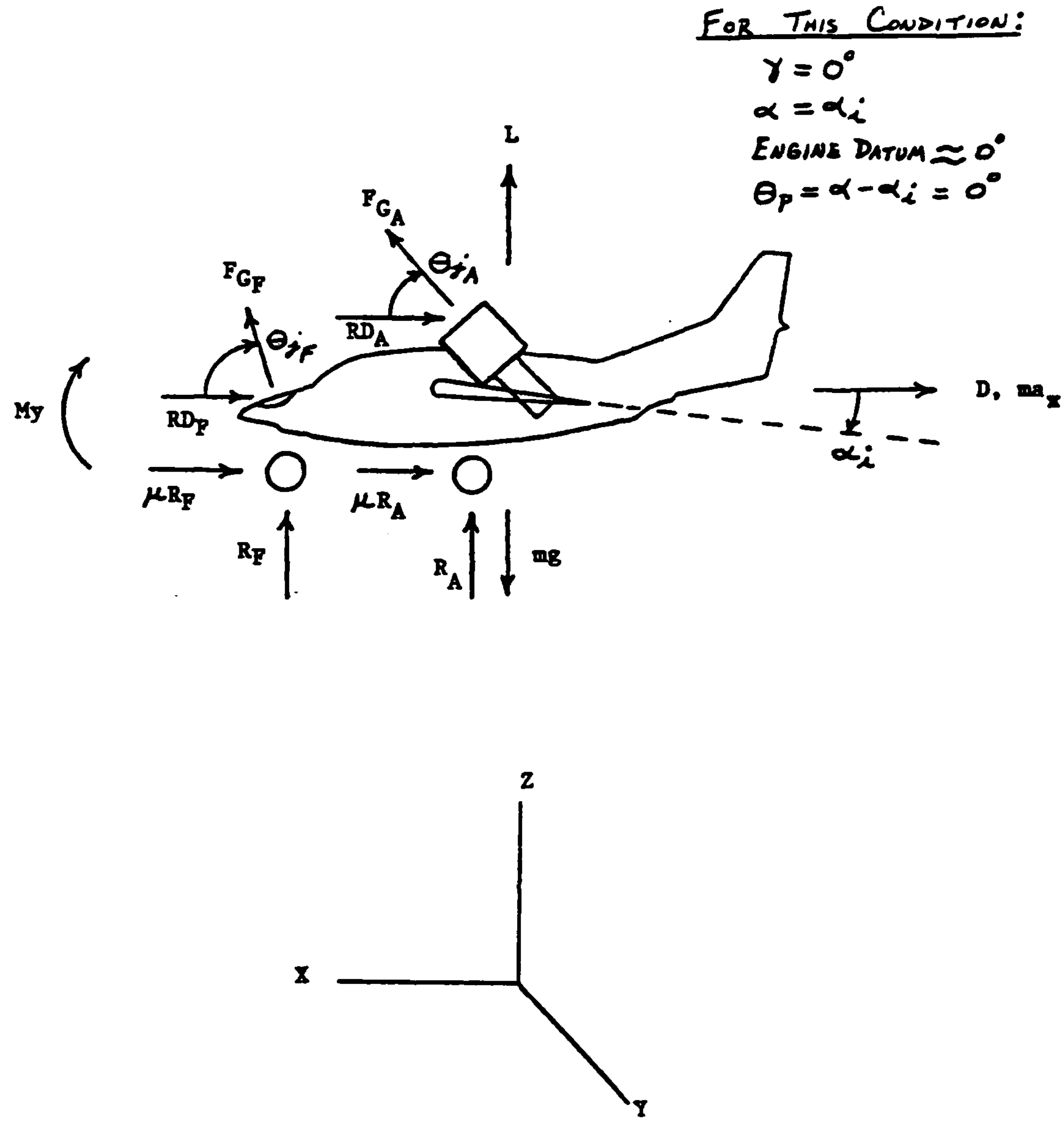


FIGURE 1. Force Diagram (Ground Run)

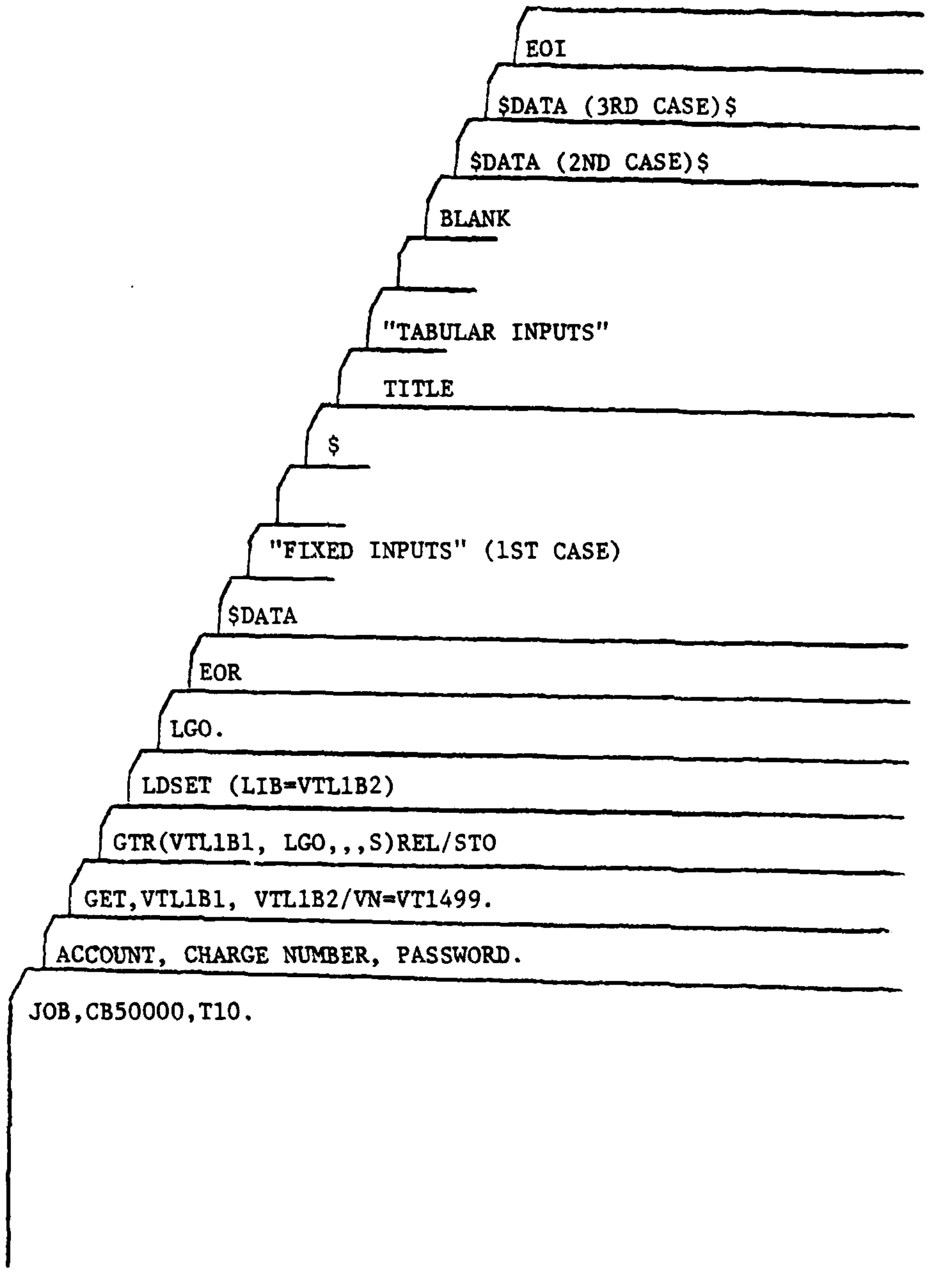


FIGURE 2. Data Input Deck Structure

TABLE I

FIXED INPUT VARIABLE LIST

Variable Name	Description	Units	Default Value
ALPHA	Initial angle of attack (for C_L-α curve)	deg.	-----
ALPHAAR	Pitch angle after rotation	deg.	0.0
ALPHARR	Pitch up rotation rate	deg/sec	0.0
ALPHMIN	Maximum angle of attack before auto rotation control attempted	deg	0.0
ALT	Field altitude	m. or ft.	0.0
ANGFRL	Initial fuselage reference line angle (for engine reference)	deg	0.0
DELMAX	Maximum control surface deflection angle (maximum δ_e)	deg	-----
DELTAE	Initial control surface deflection angle	deg	0.0
DISTMAX	Abort run if this takeoff distance is exceeded	m. or ft.	10^6
DT	Time integration increment	sec.	.05
EDATUM	Engine datum (with respect to fuselage reference line)	deg	0.0
EFAILA	If ≠ zero, failure of one engine assumed at liftoff	-----	0.0
E1	If ≠ zero, two thrust vectors are present	-----	0.0
FLDTEMP	Field ambient temperature	°C or °F	59.
IPRINT	If ≠ zero, a time history of parameters will be printed	-----	0
IYYCG	Constant aircraft moment of inertia	kg-m^2 or slug-ft^2	-----
LR	Vertical distance from center of gravity (CG) to ground	m. or ft.	-----
LRA	Horizontaol distance from CG to aft wheel	m. or ft.	-----
LRF	Horizontal distance from CG to fore wheel	m. or ft.	-----
MAC	Mean aerodynamic chord	m. or ft.	-----
METRIC	If ≠ zero, metric units present	-----	0
MU	Wheel friction coefficient	-----	.02
NEA	Number of lift cruise engines	-----	1.
NEF	Number of lift engines	-----	1.
NODOEI	If ≠ zero, tabular engine out drag present	-----	0
NOEDRAG	If ≠ zero, tabular throttle dependent drag present (includes ram drag)	-----	0
NOSTORE	If ≠ zero, tabular external store drag data present	-----	0
NPRINT	If ≠ zero, tabular input will be printed	-----	0
OEILOSF	Lift engine scale down when OEI occurs	-----	.0
OEITAB	If ≠ zero, OEI at liftoff assumed with tabular OEI data present	-----	0.
PHI1	Initial lift engine jet nozzle angle (with respect to engine datum)	deg	0.0
PHI1AR	Lift engine nozzle angle after rotation	deg	0.0

TABLE I
(Continued)

FIXED INPUT VARIABLE LIST

Variable Name	Description	Units	Default Value
PHI1RR	Lift engine nozzle rotation rate	deg/sec	0.0
PHI2	Initial lift cruise engine jet nozzle angle (with respect to engine datum)	deg	0.0
PHI2AR	Lift cruise engine nozzle angle after rotation	deg	0.0
PHI2RR	Lift cruise nozzle rotation rate	deg/sec	0.0
PHONY	If ≠ zero, instantaneous R/C is calculated	-----	0.0
PS	Pitch up aircraft speed	m/sec or ft/sec	0.0
PSINT	If ≠ zero, and RUN ≠ zero, will give interpolated pitch up speed	-----	0.0
RAM	Ram drag scale factor for OEI; if default value selected, tabular OEI ram drag is used	-----	1.
RDSCALE	Tabular ram drag scale factor	-----	1.
REVTIME	Engine speed up time	sec	0.0
ROC	If ≠ zero, dynamic liftoff characteristics at this R/C will be given	m/sec or ft/sec	0.0
RS	Airspeed which initiation of lift cruise engine nozzle rotation occurs	m/sec or ft/sec	0.0
RSF	Airspeed which initiation of lift engine nozzle rotation occurs	m/sec or ft/sec	0.0
RUN	IF ≠ zero, STO distance to determine interpolated TOGW	m or ft	0.0
S	Wing area	m^2 of ft^2	-----
SEGMENT	If default value used, no pitching moment estimation is entered in calculations; if = 1, auto rotation included in estimates; and if = 2, elevator correction to auto rotation applied	-----	0
SINKD	If ≠ zero, this is bow distance for sink calculations	m or ft	0.0
VSTARR	Maximum velocity before which a constant linear extrapolation of aerodynamic tabular data occurs	m/sec or ft/sec	0.0
W	Aircraft mass	kg or lb	-----
WOD	Wind over deck (+ for headwind, - for tailwind)	m/sec or ft/sec	0.0
XCG	Constant horizontal distance from CG to main gear wheel (presumes no tabular CG locations inputting)	m or ft	-----
YCG	Constant vertical distance from CG to main gear wheel (supposes no tabular CG locations inputted)	m or ft	-----

APPENDIX A

TABULAR INPUTS

A list of the tabular data inputs is presented in graphical form. An example of this data in numerical form can be found in Appendix B and reference (a).

Table Reference No. 13

$C_D = f(C_L, \delta_e, \theta_j, V_x)$

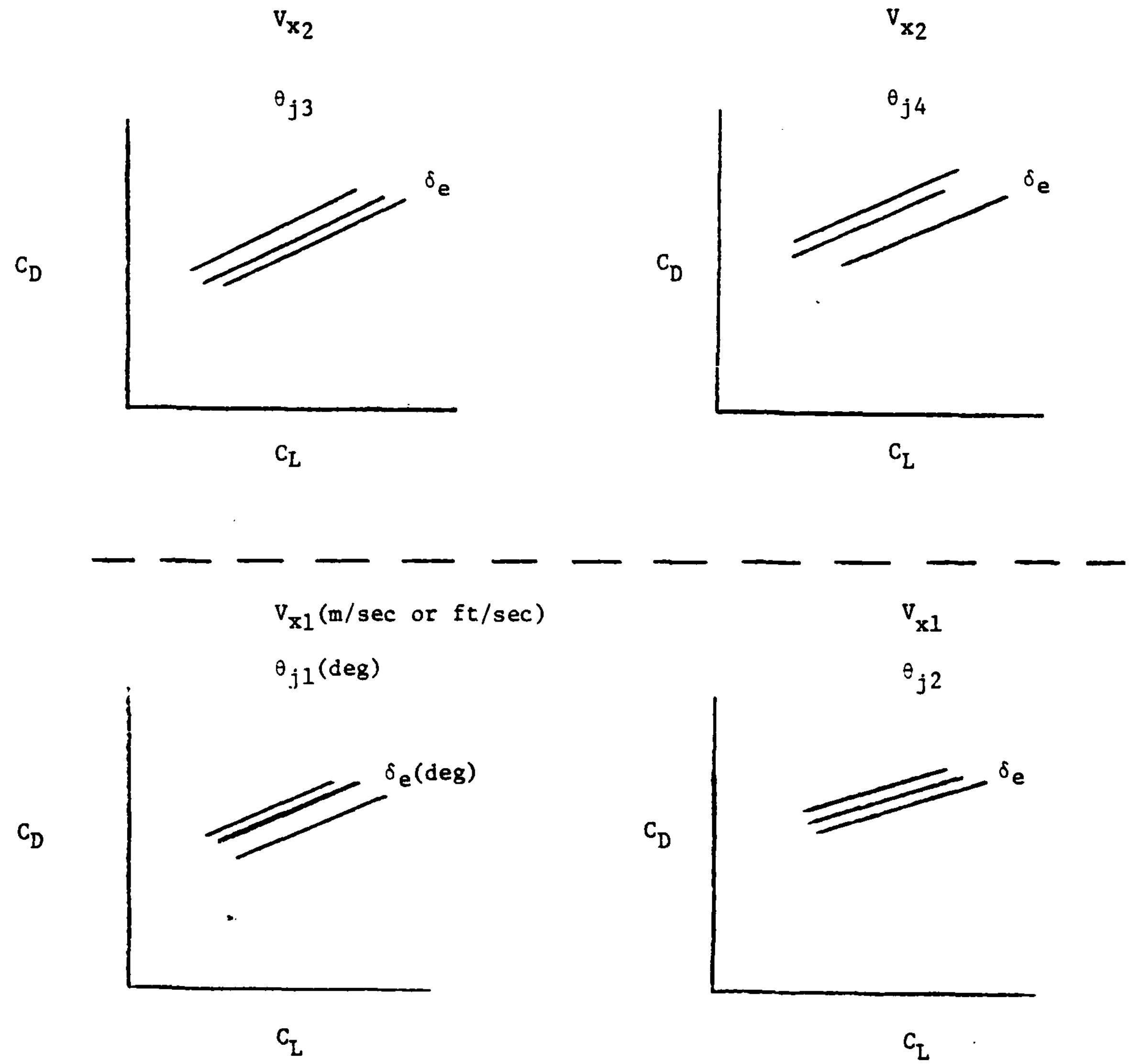


FIGURE A1. Drag Coefficient Tabular Input

Table Reference No. 14
$C_L = f(\alpha, \delta_e, \theta_j, Vx)$

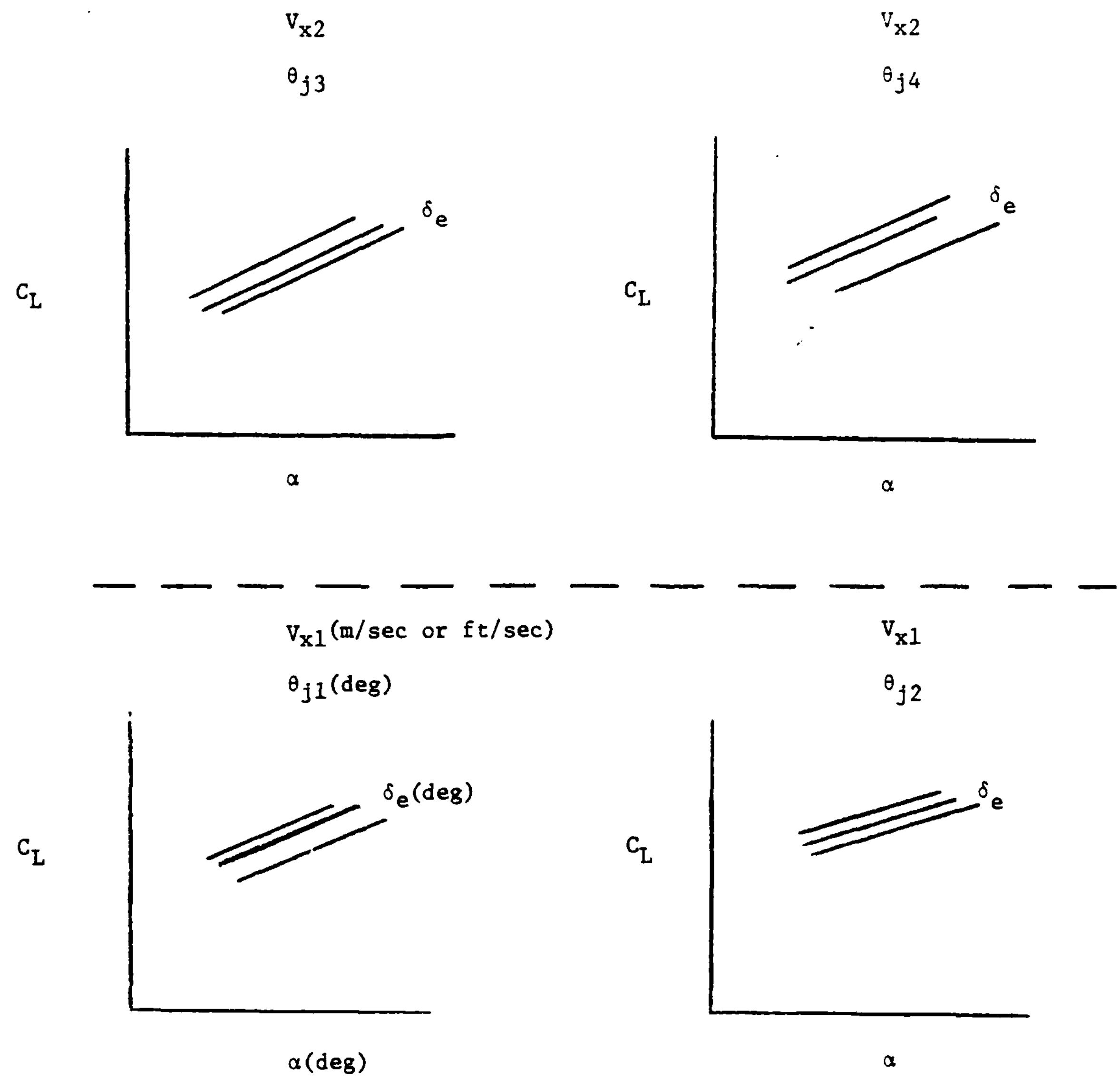
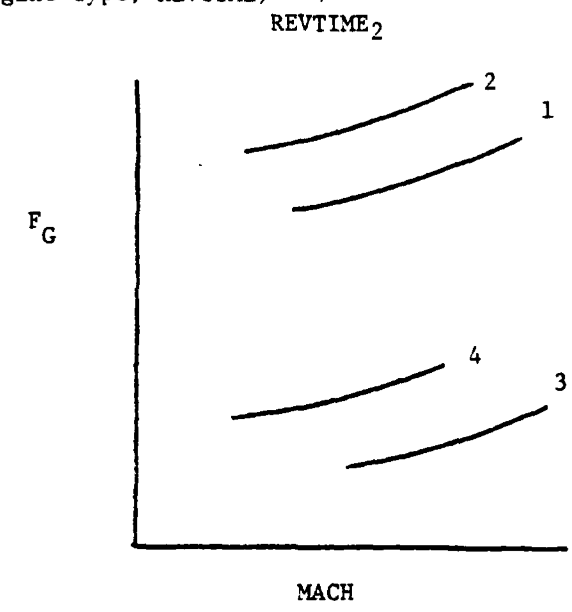


FIGURE A2. Lift Coefficient Tabular Input

Table Reference No. 10
F_G=f(MACH, Engine Type, REVTIME)



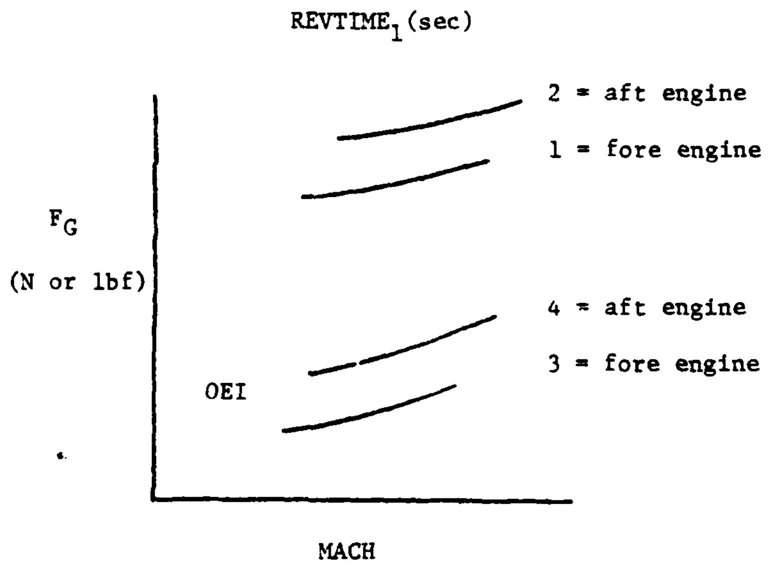


FIGURE A3. Gross Thrust Tabular Input

Table Reference No. 11

$W_F = f$(MACH, Engine Type, REVTIME)

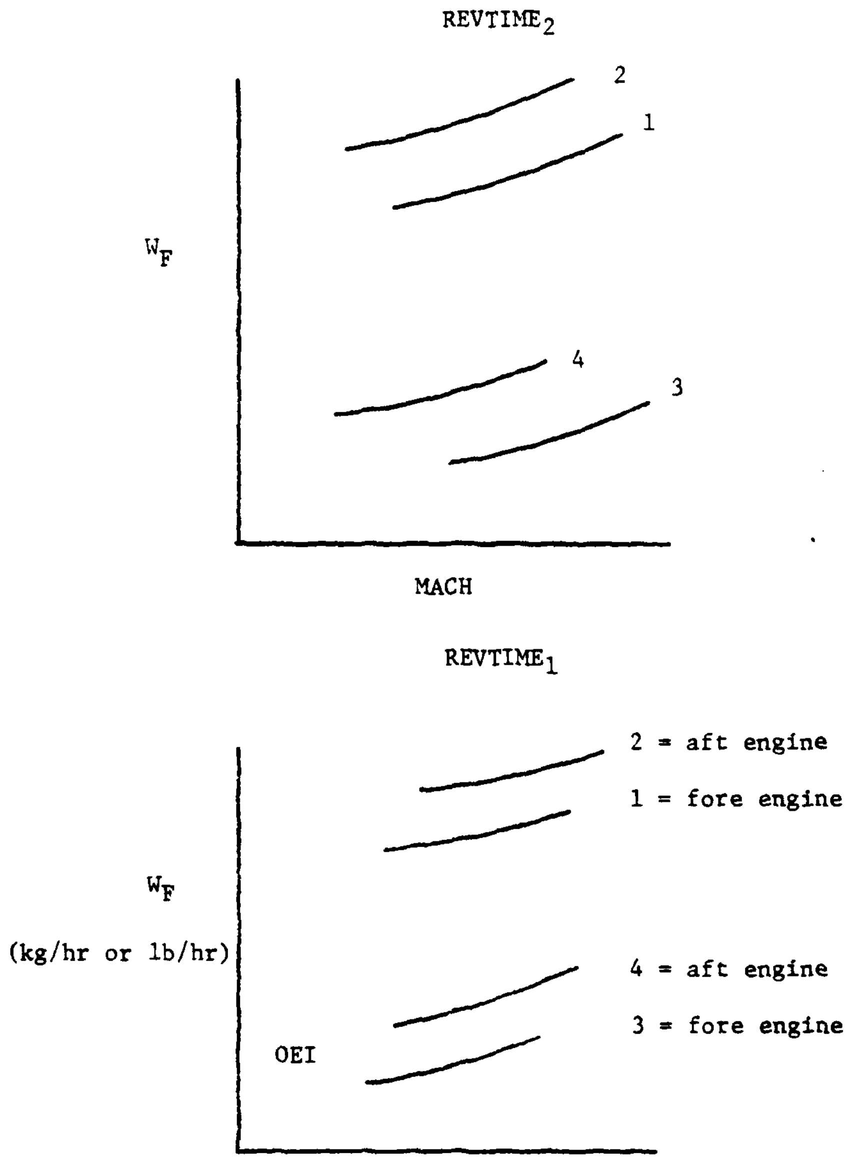


FIGURE A4. Fuel Flow Tabular Input

Table Reference No. 17
$C_M = f(C_L, \delta_e, \theta_j, V_x)$

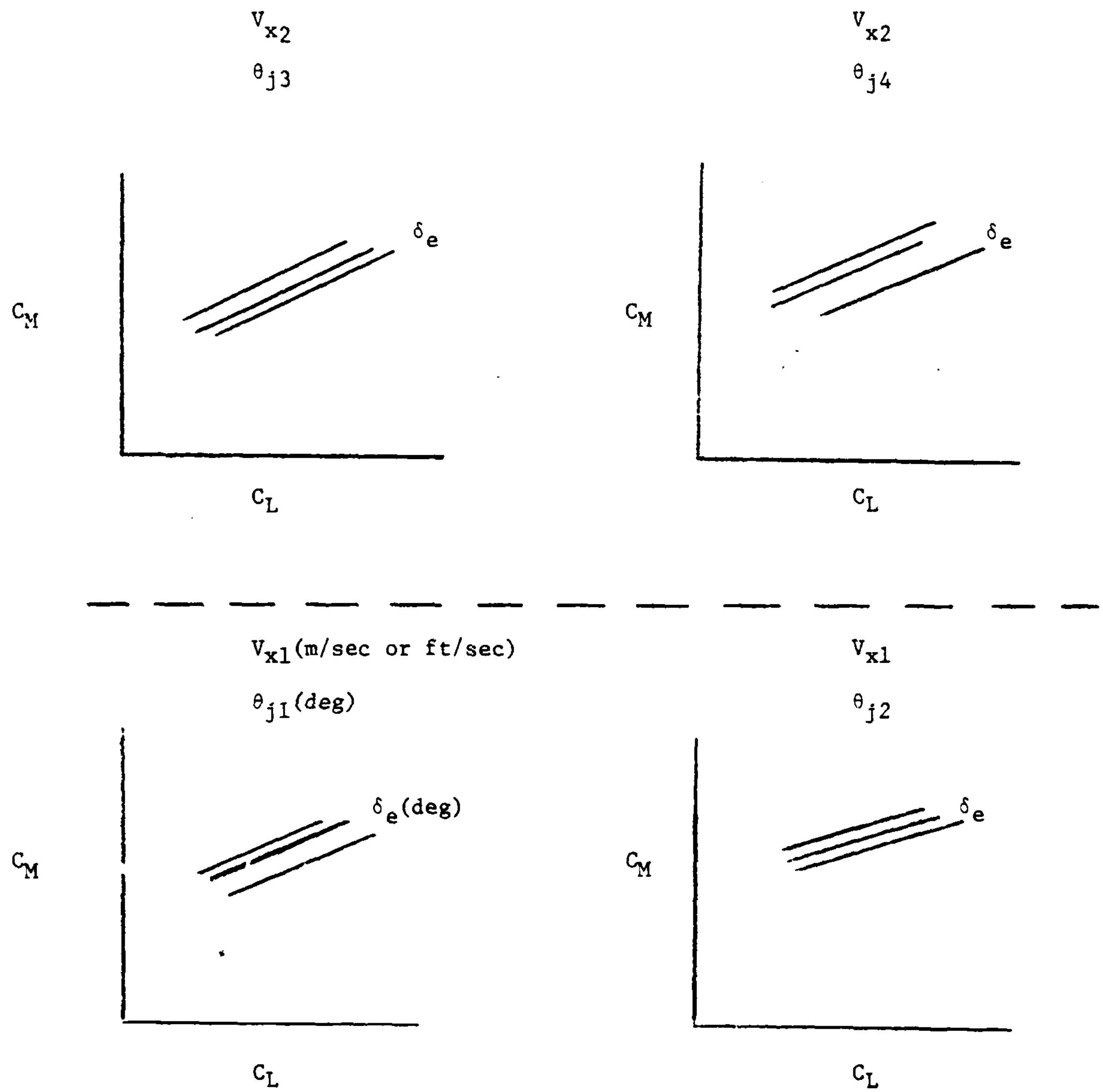


FIGURE A5. Moment Coefficient Tabular Input

Table Reference No. 30
$I_{yy_{CG}} = f(m)$

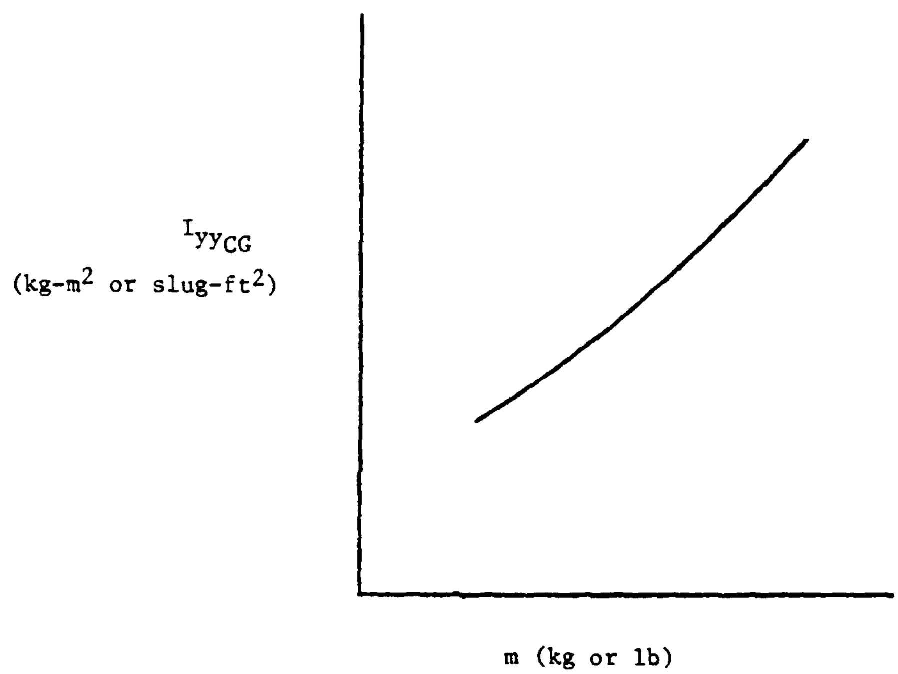


FIGURE A6. Moment of Inertia Tabular Input

Table Reference No. 15

$\frac{\Delta L}{F_G}, \frac{\Delta D}{F_G} = f(\theta j)$

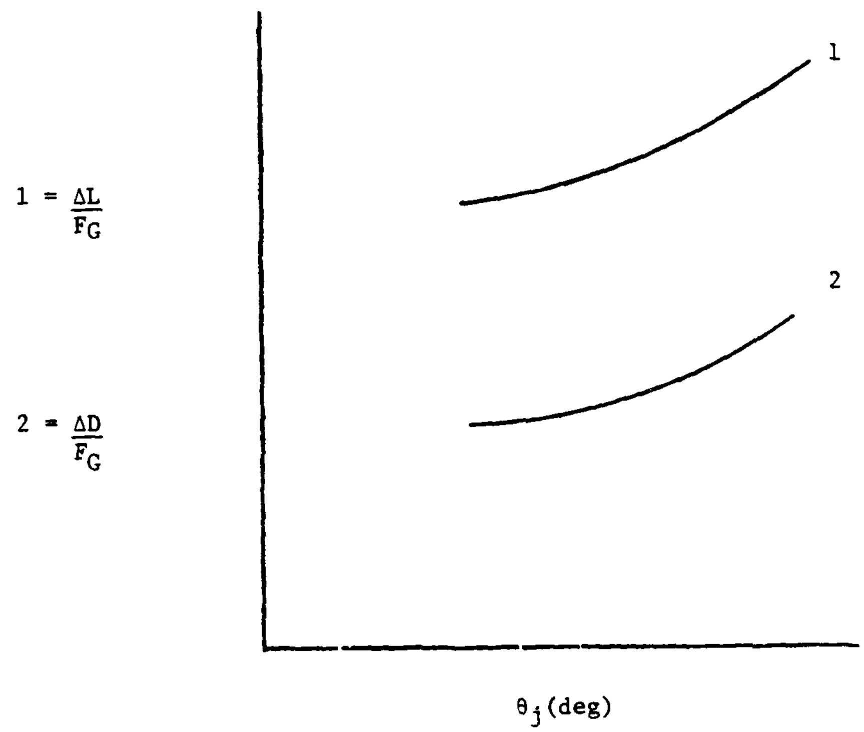


FIGURE A7. Induced Lift and Drag Tabular Input

Table Reference No. 33
D = f(MACH, Drag Type)

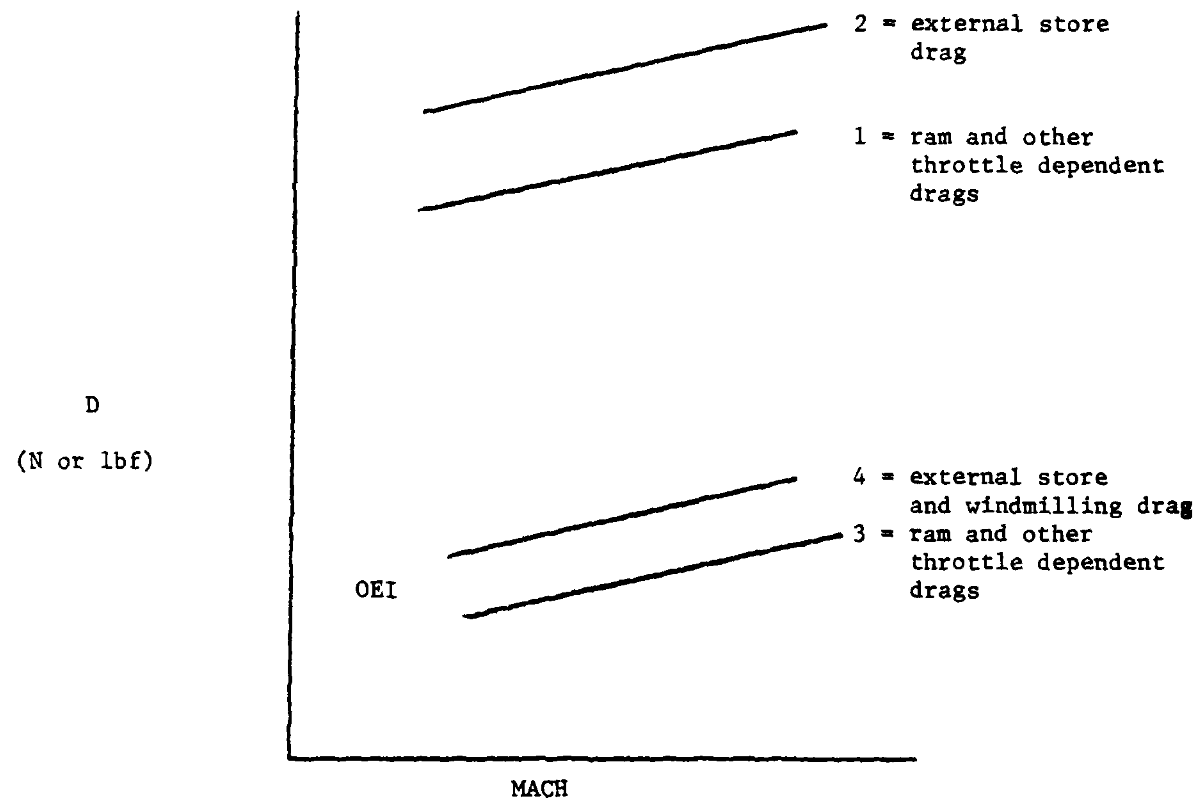


FIGURE A8. External Store and Ram Drag Tabular Input

Reference Table No. 31
X_{CG}, Y_{CG} = f(m)

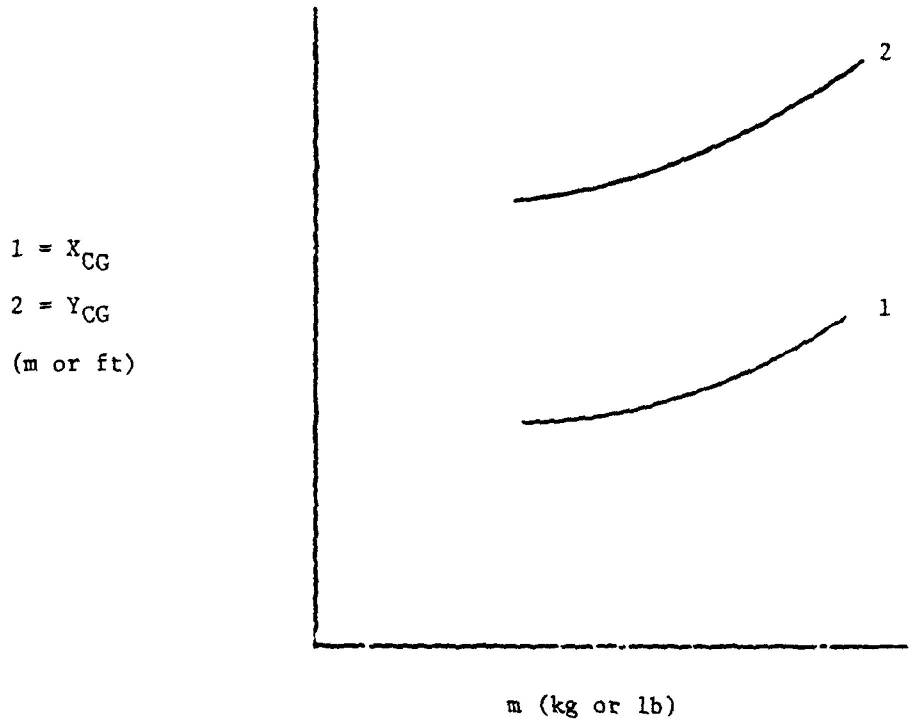


FIGURE A9. Center of Gravity Location Tabular Input

Reference Table No. 12

$F_{G_{Loss}}/F_{G_{(\text{No nozzle deflection})}} = f(\theta_j, \text{ENGINE TYPE})$

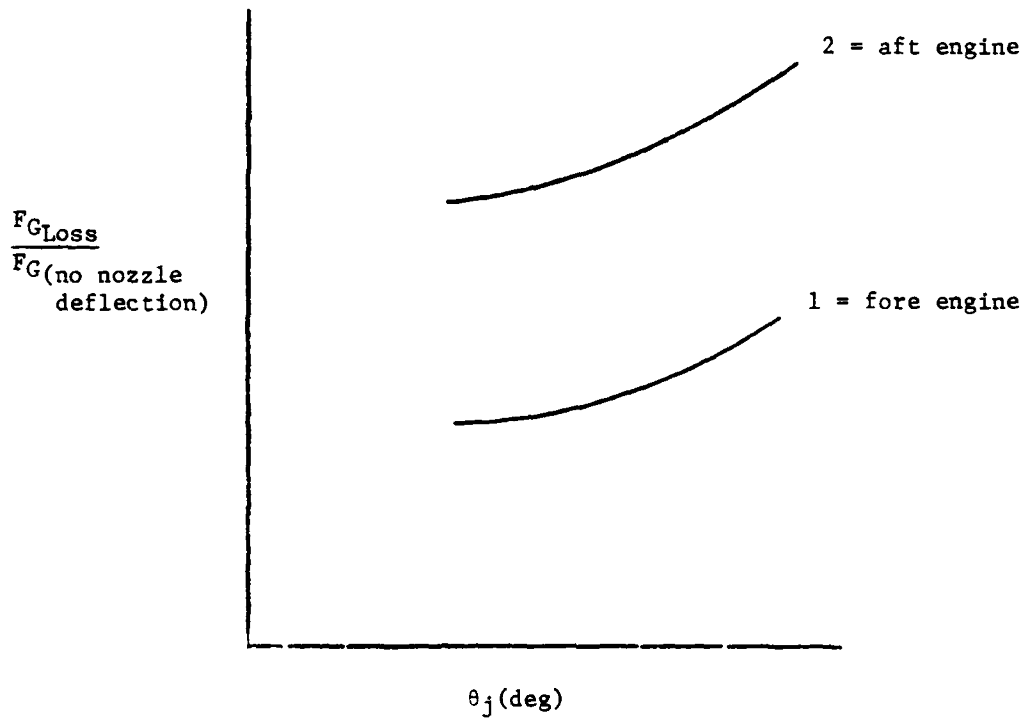


FIGURE A10. Gross Thrust Loss Tabular Input

APPENDIX B

EXAMPLE CASE

An example card deck and printout is presented. The required programs, supporting subroutines, and control cards to execute this program with any charge number under the NAVAIRDEVCEN KRONOS 2.1 operating system is illustrated. An explanation of the sample case is also presented.

Example Case

Card Deck

An example card deck listing, similar in form to the arrangement illustrated in Figure 2, is presented. As shown, three cases are to be run and they require less than 50,000 octal central memory core and ten octal seconds central processor unit time.

For the first case an aircraft, with a wing area of 430 square feet, is initially resting on the ground with a wing angle of attack and fuselage reference line at negative five degrees from the horizontal. The lift engine nozzle angle is set at 45 degrees while the cruise engine nozzle angle is set at five degrees. Both nozzle angles are with respect to the fuselage reference line (or to the engine datum line for the cruise engines). On examining the initial case's fixed inputs, at an airspeed of 66 ft/sec the vehicle will rotate its lift engine nozzles to 75 degrees at a rate of 15 degrees/sec and it will rotate its cruise engine nozzles to 54 degrees also at 15 deg/sec. At an airspeed of 80 ft/sec the aircraft will pitch up to 13 degrees at 5.9 deg/sec. This vehicle weighs 40,000 lbs. Additional fixed inputs state that the vehicle has a tabular ram drag input (NOEDRAG), two thrust vectors are present (E1), and the analysis is for a 90°F day (FLDTEMP). The default value of METRIC causes English units to be used. If the deck run exceeds 750 feet the case runs will be terminated. The variables RUN and PSINT are used, in combination, to interpolate or extrapolate the case values of PS. This is to determine the appropriate value of this variable for a deck run of 400 feet. The variable SINKD specifies the carrier bow length and signals for sink calculations to be done for this case. The IPRINT parameter indicates that a time history of pertinent parameters for this specific case will be printed.

Example tabular data includes C_L versus α, drag polars, engine thrusts, ram drag, and induced aerodynamics. The table numbers used for each table are predetermined as outlined in Appendix A. Notice that omitted input, such as fuel flow data, is assumed to be zero by the program.

The second and third fixed input cases switch off the sink calculations and the time history printout and change the lift and cruise nozzle rotation initiation airspeeds. These additional cases also change the airspeed in which the vehicle begins pitching up. Note that a maximum of one card is permitted for each additional case. For the additional cases a value for ROC causes dynamic rate of climb calculations to be made.

```
TEST.C950009.T10.
ACCOUNT.CHARGE NUMBER.PASSWORD.
GET.VTLIB1.VTLIB2/UN=VT1499.
GTR(VTLIB1)LGO...S1FEL/STO
LDSET(LIB=VTLIB2)
LGO.
END OF RECORD
 $DATA ALPHA=-5.S=430.PHI2=5.PHI2AR=54.PHI2RR=15.ALPHARR=5.9.ALPHAAR=13.
 ANGFRL=-5.NOEDRAG=1.E1=1.FLDTEMP=90.PHI1=45.DISTMAX=750.PHI1AR=75.
 PHI1RR=15.RS=66.PS=80.RSF=66.W=40000.IPRINT=1.PSINT=1.HUM=400.
 SINKP=200.NPRINT=1.
 $
                EXAMPLE CASE
   14      EXAMPLE CL VS. ALPHA TABLE
VEL    1   0.
PHI2   3   0.          50.         40.
DLTE   1   0.
ALPH   9   -8.         -4.         0.          4.          8.          12.         16.
           20.
CL     9   -.48        -.175       .175        .45         .78         1.04        1.375
           1.49
DLTE   1   0.
CL     8   -.13        .15         .42         .69         .96         1.2         1.45
           1.57
DLTE   1   0.
CL     8   .34         .59         .86         1.11        1.34        1.47        1.56
           1.55
EOT
   13        EXAMPLE DRAG POLARS (POWER OFF) TABLE
DUMM   1   0.
PHI2   1   0.
DLTE   1   0.
CL     6   -.4         0.          .4          .8          1.2         1.4
CD     6   .34159      .02938      .03742      .06533      .17463      .22905
EOT
   10      EXAMPLE ENGINE THRUST TABLE
DUMM   1   0.
TIME   1   0.
SET    4   1.          2.          3.          4.
MACH   5   0.          .05         .1          .15         .2
FG     5   13755.      14044.      14332.      14875.      15417.
FG     5   29437.      29988.      30538.      31606.      32673.
FG     5   10994.      11330.      11565.      12342.      13018.
FG     5   23734.      24346.      24958.      26161.      27363.
EOT
   33      EXAMPLE RAM DRAG TABLE
DUMM   1   0.
TIME   1   0.
SET    2   1.          3.
MACH   5   0.          .05         .1          .15         .2
RAMD   5   0.          3936.       9123.       12240.      16021.
RAMD   5   0.          3862.       7931.       12023.      16486.
EOT
   15     EXAMPLE INDUCED AERODYNAMICS TABLE
DUMM   1   0.
Y      1   0.
SET    2   1.          2.
PHI2   3   0.          50.         100.
DFL    3   .32         .12         .214
PHI2   3   27.3        44.5        84.7
DFL    3   -.358       -.031       .042
EOT
```

```
BLANK CARD
 $DATA SINKD=0,ROC=1.67,PS=90,RS=76,RSF=76,IPRINT=0$
 $DATA SINKD=0,ROC=1.67,PS=100,RS=84,RSF=84,IPRINT=0$
NOTE: MAXIMUM OF ONE CARD FOR EACH ADDITIONAL CASE.
END OF INFORMATION
```

Printout

The computer printout of the previously mentioned card deck is now presented. On the second page of the printout a loader map illustrating all the supporting subroutines and their individual core requirements is presented. On the bottom of this page the fixed inputs for the first case are shown. The variable NPRINT enables the tabular input to be printed in full on the following five successive pages. This data, in turn, is followed by a summary of core requirements for each table. Next, the output data begins with a time history of the first of three sample cases. Each element is a successive point in a time variance of parameters. These parameters include:

W - Aircraft mass (kg or lb) (English units for this example)
V - Vehicle airspeed (m/sec or ft/sec)
L - Lift force (N or lbf)
DR - Drag force (N or lbf)
RF - Nose gear reaction force (N or lbf) (not used in this example)
EDRG - Engine related drags (N or lbf)
FN2 - Cruise engines gross thrust (N or lbf)
PH2I - Cruise engine nozzle angle (deg)
ALPHA - Angle of attack (deg)
T - Time (sec)
D2 - Distance (m or ft)
BALANCE - Lifting forces minus weight (N or lbf)
PH1I - Lift engine nozzle angle (deg)
FN1 - Lift engines gross thrust (N or lbf)
C_L - Lift coefficient
C_D - Drag coefficient
C_M - Pitching moment coefficient (not used this example)
A/G - Normalized longitudinal acceleration
RHO - Atmospheric density (kg/m^3 or slug/ft^3)
DELTAE - Control surface deflection angle (deg) (not used this example)

The following parameters were not printed in this listing because they were not used for this example case.

DALPHAD - Angle of attack increment caused by auto rotation (deg)
IYYCG - Lateral aircraft moment of inertia (kg-m^2 or slug-ft^2)
XCG - Horizontal distance from CG to main gear wheel (m or ft)
YCG - Vertical distance from CG to main gear wheel (m or ft)
N - Iteration number for elevator correction to counter auto rotation
RF - Nose gear reaction force during iteration (N or lbf)
DELTAE - Control surface deflection during iteration (deg)

The following parameters are printed if dynamic rate of climb calculations are made. These parameters are printed only after liftoff.

ROC - Dynamic rate of climb (m/sec or ft/sec)
ACCY - Vertical dynamic acceleration (m/sec^2 or ft/sec^2)

Finally, the following parameters (which are used for this example case) are printed after liftoff when sink calculations are made.

ASINK - Sink acceleration (m/sec^2 or ft/sec^2) (sink calculations cease when this value becomes zero)

VSINK - Sink velocity (m/sec or ft/sec)
DSINK - Sink distance (m or ft)

The time history printout concludes on the following page of output. After liftoff the appropriate sink parameters are shown. After the time history the final liftoff parameters are presented. These parameters include maximum sink height and time, distance, and airspeed at that sink height condition. Other characteristics at this condition include lift, drag, horizontal acceleration, gross thrust, and vertical gross thrust.

Since the IPRINT was switched off for the two additional cases no time history is printed. These cases were done with dynamic rate of climb calculations so instead of sink parameters the prescribed rate of climb along with the vertical height when this rate of climb is achieved is printed.

After all the cases are determined an extrapolation is made to estimate the pitchup airspeed and horizontal acceleration for a deck run of 400 feet.

KRONOS 2.1.1-107   81/03/30.
OPERATING SYSTEM
JOB ORIGIN = EI200.

```
TTTTTTTTTTTT  EEEEEEEEEEEE   SSSSSSSSSS   TTTTTTTTTTTT   AAAAAAAAAA   VV          VV  IIIIIIIIIIII
TTTTTTTTTTTT  EEEEEEEEEEEE  SSSSSSSSSSSS  TTTTTTTTTTTT  AAAAAAAAAAAA  VV          VV  IIIIIIIIIIII
     TT       EE            SS         S       TT       AA        AA  VV          VV       II
     TT       EE            SS                 TT       AA        AA   VV        VV        II
     TT       EE            SS                 TT       AA        AA   VV        VV        II
     TT       EE            SS                 TT       AA        AA   VV        VV        II
     TT       EE            SS                 TT       AA        AA    VV      VV         II
     TT       EEEEEEEE      SSSSSSSSSSS        TT       AA        AA    VV      VV         II
     TT       EEEEEEEE       SSSSSSSSSSS       TT       AAAAAAAAAAAA    VV      VV         II
     TT       EE                      SS       TT       AAAAAAAAAAAA     VV    VV          II
     TT       EE                      SS       TT       AA        AA     VV    VV          II
     TT       EE                      SS       TT       AA        AA     VV    VV          II
     TT       EE                      SS       TT       AA        AA      V    V           II
     TT       EE            S         SS       TT       AA        AA      VVVV             II
     TT       EEEEEEEEEEEE  SSSSSSSSSSSS       TT       AA        AA      VVVV        IIIIIIIIIIII
     TT       EEEEEEEEEEEE   SSSSSSSSSS        TT       AA        AA       VV         IIIIIIIIIIII
```

81/09/29. 14.54.07.

```
LOAD MAP - STO                                      CYBER LOADER 1.4-485       81/09/28. 14.54.01.       PAGE      1

FWA OF THE LOAD           111
LWA+1 OF THE LOAD       32715

TRANSFER ADDRESS -- STO              6206

PROGRAM ENTRY POINTS --       STO              6206
```

PROGRAM AND BLOCK ASSIGNMENTS.

BLOCK	ADDRESS	LENGTH	FILE	DATE	PROCSSR	VER	LEVEL	HARDWARE	COMMENTS
/PRINT/	111	1							
STO	112	11662	LGO	78/01/0	FTN	3.0	P380	6666	
/TSIZE/	11774	5671							
TAKES	17665	1353	UL-VTLIB2	77/03/0	FTN	3.0	P380	6666	
SPLN01	21240	504	UL-VTLIB2	77/10/1	FTN	3.0	P380	6666	
UPDATE	21744	135	UL-VTLIB2	75/07/1	FTN	3.0	P380	6666	
ATMOS	22101	213	UL-VTLIB2	76/10/2	FTN	3.0	P380	6666	
SNEST	22314	110	UL-VTLIB2	75/06/1	FTN	3.0	P380	6666	
SYSTEM$	22424	1035	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
ACGOER$	23461	13	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
BACKSP$	23474	334	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
IFENDF$	24030	57	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
INPUTC$	24107	131	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
INPUTN$	24240	1171	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
OUTPTC$	25431	74	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
REWINM$	25525	52	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
EXPE	25577	44	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
SINCOSE	25643	55	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
SQRTE	25720	22	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
XTOYE	25742	7	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
KODER$	25751	1422	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
KRAKER$	27373	1525	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
ALNLOGE	31120	37	SL-FTN3LIB	80/09/02	COMPASS	3.6	485		
GETBA	31157	17	SL-FTN3LIB	80/09/0					
SIO$	31176	1517	SL-FTN3LIB	80/09/0					

```
        .106 CP SECONDS              467008 CM STORAGE USED                        11 TABLE MOVES


$DATA ALPHA=-5.S=430.PHI2=5.PHI2AR=54.PHI2RR=15.ALPHARR=5.9.ALPHAAR=13.
ANGFRL=-5.NOEDRAG=1.FI=1.FLOTEMP=90.PHI1=45.DISTMAX=750.PHI1AR=75.
PHI1RR=15.HS=66.PS=80.RSF=66.W=40000.IPRINT=1.PSINT=1.RUN=400.
SINAD=200.PPRINT=1.
$
```

```
14     EXAMPLE CL VS. ALPHA TABLE


VEL  =    .0
PHI2=     .0            CLTE=    .0
                        ALPH    -.80000E+01   -.40000E+01    .0            .40000E+01    .80000E+01    .12000E+02    .16000E+02    .20000E+02
                        CL      -.48000E+00   -.17500E+00    .17500E+00    .45000E+00    .78000E+00    .10400E+01    .13750E+01    .14900E+01
VEL  =    .0
PHI2=     .50000E+02    CLTE=    .0
                        ALPH    -.80000E+01   -.40000E+01    .0            .40000E+01    .80000E+01    .12000E+02    .16000E+02    .20000E+02
                        CL      -.13000E+00    .15000E+00    .42000F+00    .69000E+00    .96000E+00    .12000E+01    .14500E+01    .15700E+01
VEL  =    .0
PHI2=     .90000E+02    CLTE=    .0
                        ALPH    -.80000E+01   -.40000E+01    .0            .40000E+01    .80000E+01    .12000E+02    .16000E+02    .20000E+02
                        CL       .34000E+00    .59000E+00    .86000F+00    .11100E+01    .13400E+01    .14700E+01    .15600E+01    .15500E+01
```

```
 13      EXAMPLE DRAG POLARS (POWER OFF) TABLE


DUMM=    .0
PHI2=    .0          CLTE=    .0
                     CL      -.40000E+00   .0            .40000E+00   .80000E+00   .12000E+01   .14000E+01
                     CD       .41590E-01   .29380E-01    .37420E-01   .65330E-01   .17463E+00   .22905E+00
```

```
10     EXAMPLE ENGINE THRUST TABLE



DUMM=   .0
TIME=   .0          SET =   .10000E+01
                    MACH    .0            .50000E-01    .10000E+00    .15000E+00    .20000E+00
                    FG      .13755E+05    .14044E+05    .14332E+05    .14875E+05    .15417E+05
DUMM=   .0
TIME=   .0          SET =   .20000E+01
                    MACH    .0            .50000E-01    .10000E+00    .15000E+00    .20000E+00
                    FG      .29437E+05    .29988E+05    .30538E+05    .31606E+05    .32673E+05
DUMM=   .0
TIME=   .0          SET =   .30000E+01
                    MACH    .0            .50000E-01    .10000E+00    .15000E+00    .20000E+00
                    FG      .10994E+05    .11330E+05    .11665E+05    .12342E+05    .13018E+05
DUMM=   .0
TIME=   .0          SET =   .40000E+01
                    MACH    .0            .50000E-01    .10000E+00    .15000E+00    .20000E+00
                    FG      .23774E+05    .24346E+05    .24958E+05    .26161E+05    .27363E+05
```

```
  33      EXAMPLE RAM DRAG TABLE


DUMM=    .0
TIME=    .0          SET =    .10000E+01
                     MACH     .0          .50000E-01   .10000E+00   .15000E+00   .20000E+00
                     RAMD     .0          .39860E+04   .81230E+04   .12240E+05   .16621E+05
DUMM=    .0
TIME=    .0          SET =    .30000E+01
                     MACH     .0          .50000E-01   .10000E+00   .15000E+00   .20000E+00
                     RAMD     .0          .38620E+04   .79310E+04   .12023E+05   .16486E+05
```

15 EXAMPLE INDUCED AERODYNAMICS TABLE

```
DUMM=   .0
Y   =   .0          SET =   .10000E+01
                    PHI2    .0              .50000E+02   .10000E+03
                    CEL     .20000E-01      .12000E+00   .21400E+00
DUMM=   .0
Y   =   .0          SET =   .20000E+01
                    PHI2    .20300E+02      .44500E+02   .84700E+02
                    CEL    -.58000E-01     -.31000E-01   .42000E-01
```

TABLE DATA INPUT SUMMARY 5 TABLES

TABLE NUMBER	REFERENCE NUMBER	ARRAY LOCATION
1	10.	217.
2	13.	178.
3	14.	61.
4	15.	373.
5	33.	316.

DATA STORAGE ALLOCATION 3000
DATA STORAGE NOT USED 2582

```
VSTOL TAKEOFF PERFORMANCE ESTIMATION
------------------------------------

     EXAMPLE CASE

TOGW =            .40000E+05

INITIAL GROUND RUN PARAMETERS

$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
W,V,L,DR,RF,EDRA,FN2
PH2I,ALPHA,T,D2,BALANCE,PHII,FN1
CL,CD,CM,A/G,RHO,DELTAE
ALPHA,DALPHAD,IYYCG,XCG,YCG
N,RF,DELTAE
ROC,ACCY
ASINK,VSINK,DSINK
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
  .40000E+05     .84870E+01     .13032E+04    -.24404E+04          IIIII     .57810E+03     .29518E+05
  .50000E+01    -.50000E+01     .25000E+00     .15292E+01     .29828E+05     .45000E+02     .13798E+05
 -.23089E+00     .33947E-01          IIIII     .10483E+01     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .16882E+02     .12832E+04    -.18718E+04          IIIII     .11518E+04     .29599E+05
  .50000E+01    -.50000E+01     .50000E+00     .51215E+01     .29821E+05     .45000E+02     .13840E+05
 -.23089E+00     .33947E-01          IIIII     .10369E+01     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .25186E+02     .12479E+04    -.13042E+04          IIIII     .17221E+04     .29679E+05
  .50000E+01    -.50000E+01     .75000E+00     .10797E+02     .29829E+05     .45000E+02     .13882E+05
 -.23089E+00     .33947E-01          IIIII     .10255E+01     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .33398E+02     .11980E+04    -.73685E+03          IIIII     .22900E+04     .29757E+05
  .50000E+01    -.50000E+01     .10000E+01     .18532E+02     .29852E+05     .45000E+02     .13923E+05
 -.23089E+00     .33947E-01          IIIII     .10141E+01     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .41518E+02     .11339E+04    -.16891E+03          IIIII     .28563E+04     .29835E+05
  .50000E+01    -.50000E+01     .12500E+01     .28304E+02     .29890E+05     .45000E+02     .13964E+05
 -.23089E+00     .33947E-01          IIIII     .10026E+01     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .49545E+02     .10560E+04     .40027E+03          IIIII     .34216E+04     .29912E+05
  .50000E+01    -.50000E+01     .15000E+01     .40039E+02     .29942E+05     .45000E+02     .14004E+05
 -.23089E+00     .33947E-01          IIIII     .99102E+00     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .57479E+02     .96502E+03     .97133E+03          IIIII     .39869E+04     .29988E+05
  .50000E+01    -.50000E+01     .17500E+01     .53865E+02     .30008E+05     .45000E+02     .14044E+05
 -.23089E+00     .33947E-01          IIIII     .97938E+00     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .65319E+02     .86111E+03     .15422E+04          IIIII     .45496E+04     .30059E+05
  .50000E+01    -.50000E+01     .20000E+01     .69603E+02     .30088E+05     .45000E+02     .14081E+05
 -.23089E+00     .33947E-01          IIIII     .96755E+00     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .73008E+02     .11342E+04     .22311E+04          IIIII     .51025E+04     .30123E+05
  .87500E+01    -.50000E+01     .22500E+01     .47287E+02     .27135E+05     .48750E+02     .14115E+05
 -.21161E+00     .33705E-01          IIIII     .93703E+00     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .80434E+02     .14937E+04     .24017E+04          IIIII     .56376E+04     .30183E+05
  .12500E+02    -.47050E+01     .25000E+01     .10684E+03     .23913E+05     .52500E+02     .14147E+05
 -.16863E+00     .32045E-01          IIIII     .90064E+00     .22429E-02     .0
*************************************************************************************************************************************
  .40000E+05     .87525E+02     .22332E+04     .35441E+04          IIIII     .61487E+04     .30242E+05
  .16250E+02    -.32100E+01     .27500E+01     [illegible]     .19627E+05     .56250E+02     [illegible]
```

```
     -.34668E-01      .24630E-01        IIIII       .65141E+00      .22429E-02      .0
*****************************************************************************************************
      .40000E+05      .94195E+02      .31384E+04      .41654E+04        IIIII       .66297E+04      .30301E+05
      .20000E+02     -.17550E+01      .30000E+01      .15125E+03      .15293E+05      .60000E+02      .14209E+05
      .10182E+00      .29817E-01        IIIII       .79463E+00      .22429E-02      .0
*****************************************************************************************************
      .40000E+05      .10039E+03      .41758E+04      .47722E+04        IIIII       .70759E+04      .30362E+05
      .23750E+02     -.28000E+00      .32500E+01      .17589E+03      .10990E+05      .63750E+02      .14241E+05
      .23288E+00      .32481E-01        IIIII       .73048E+00      .22429E-02      .0
*****************************************************************************************************
      .40000E+05      .10604E+03      .53114E+04      .53659E+04        IIIII       .74830E+04      .30424E+05
      .27500E+02      .11950E+01      .35000E+01      .20199E+03      .67826E+04      .67500E+02      .14273E+05
      .35522E+00      .36059E-01        IIIII       .65930E+00      .22429E-02      .0
*****************************************************************************************************
SINK DATA                     .54556E+01                  .13639E+00                  .34097E-02
      .40000E+05      .11111E+03      .65542E+04      .59320E+04        IIIII       .76481E+04      .30486E+05
      .31250E+02      .26700E+01      .37500E+01      .22939E+03      .26876E+04      .71250E+02      .14305E+05
      .47494E+00      .38411E-01        IIIII       .58321E+00      .22429E-02      .0
*****************************************************************************************************
SINK DATA                     .21618E+01                  .10857E+01                  .17277E+00
        VEHICLE IS AIRBORNE
        MAXIMUM SINK DISTANCE IS          .41562E+00 FT.
        DISTANCE =            .25216E+03 FT.
        VELOCITY =            .11553E+03 FT./SEC.
        TIME =                .40000E+01 SEC.
        END WEIGHT FOR THIS PHASE IS           .40000E+05 POUNDS MASS


        LIFTOFF CHARACTERISTICS

        HORIZONTAL ACCELERATION =           .51570E+00 GIS
        LIFT =                              .76359E+04 POUNDS FORCE
        DRAG =                              .63560E+04 POUNDS FORCE
        VERTICAL GROSS THRUST =             .32863E+05 POUNDS FORCE
        HORIZONTAL GROSS THRUST =           .26984E+05 POUNDS FORCE

 NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT

 $DATA SINKD=0,ROC=1.67,PS=90,RS=76,RSF=76,IPRINT=0$
        TOGW =            .40000E+05

        INITIAL GROUND RUN PARAMETERS

        VEHICLE IS AIRBORNE
        VEHICLE HAS A R/C OF              .16700E+01 FT./SEC.
        VERTICAL HIEGHT IS            .36921E+00 FT.
        DISTANCE =            .35474E+03 FT.
        VELOCITY =            .13093E+03 FT./SEC.
        TIME =                .48000E+01 SEC.
        END WEIGHT FOR THIS PHASE IS           .40000E+05 POUNDS MASS


        LIFTOFF CHARACTERISTICS

        HORIZONTAL ACCELERATION =           .36399E+00 GIS
        LIFT =                              .11300E+05 POUNDS FORCE
        DRAG =                              .81952E+04 POUNDS FORCE
        VERTICAL GROSS THRUST =             .37272E+05 POUNDS FORCE
        HORIZONTAL GROSS THRUST =           .22635E+05 POUNDS FORCE

 NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT

 $DATA SINKD=0,ROC=1.67,PS=100,RS=86,RSF=86,IPRINT=0$
        TOGW =            .40000E+05

        INITIAL GROUND RUN PARAMETERS
```

```
VEHICLE IS AIRBORNE
VEHICLE HAS A R/C OF              .16700E+01 FT./SEC.
VERTICAL HIEGHT IS             .33656E+00 FT.
DISTANCE =             .39324E+03 FT.
VELOCITY =             .13886E+03 FT./SEC.
TIME =                 .50500E+01 SEC.
END WEIGHT FOR THIS PHASE IS             .40000E+05 POUNDS MASS

LIFTOFF CHARACTERISTICS

HORIZONTAL ACCELERATION =           .36907E+00 GIS
LIFT =                              .11639E+05 POUNDS FORCE
DRAG =                              .86602E+04 POUNDS FORCE
VERTICAL GROSS THRUST =             .36941E+05 POUNDS FORCE
HORIZONTAL GROSS THRUST =           .23423E+05 POUNDS FORCE

NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT NEXT

BY CUBIC SPLINE INTERPOLATION OF THE ABOVE DATA POINTS
              FOR A RUN OF           .40000E+03 FEET THE  PS  IS          .10320E+03 FT./SEC.
                                                AND THE HORIZONTAL ACCELERATION IS          .37630E+00 GIS
```

```
    TESTAVI.    81/09/24.KRONOS 2.1.1-SYS-C25-NADC.

.51.TEST.C450000.110.
.51.ACCOUNT.60334A..
.51. MASTER DEVICE:  6
.52.GET.VTLIB1.VTLIB2/UN=VT1499.
.57.GTR(VTLIB1.LGO...SIREL/STO
.00. EDITING COMPLETE.
.00.LDSET(LIB=VTLIB2)
.00.LGO.
.04.STOP
.04.COMPUTER UNITS AT.100  P = 1.
.04.CP        1.680 SEC.         $      0.88
.04.CM       14.323 K-S.
.04.IO        1.895 SEC.         $      0.09
.04.IM       19.838 K-S.
.04.TAPES SCHEDULED 00
.04.PACKS SCHEDULED 00
.04.SERVICE CHARGE               $      2.20
.04.                              ---------
.04.SUBTOTAL                     $      3.17
```

APPENDIX C

Program Listings

A listing of the FORTRAN source code for the main STO program and a four degree-of-freedom interpolating subroutine TAKE5 is presented.

```
PROGRAM STD          74/74    OPT=1                        FTN 4.6+428          81/09/24. 10.15.43          PAGE      1

      CSTD***
            PROGRAM STD (INPUT,OUTPUT,TAPE8,TAPE9=INPUT)
      C*****************************************************************
      C  CREATED BY KOHLS ON APRIL 1977.
      C*****************************************************************
            DIMENSION TITLE(8),ICOPY(10),RUNDATA(33),K(4),AGDATA(33),
           * PHDATA(33)
            COMMON/PRINT/NPRINT
            REAL MACH,NEF,NEA,MU,L,LW,LRA,LRF,MAC,MOMENT,LEVER,IYY,IYYCG
            INTEGER SEGMENT
            NAMELIST/DATA/ALPHA,S,DELTAE,NEF,NEA,MU,W,LR,LRA,LRF,REVTIME,DT,
           *IPRINT,NPRINT,VDATA,MAC,SEGMENT,IYYCG,XCG,YCG,EL,DELMAX,PHI1,
           *ANGFRL,FLOTEMP,ALPHMIN,PHI2,PHI2AR,PHI2RR,RS,ALPHARR,ALPHAAR,ALT,
           * RUN,EDATUM,NODRAG,NOSTORE,EFAIL,DEILOSF,NODOEI,METRIC,PS,WDD,
           *DEITAB,RSF,PHI1RR,PHI1AR,PSINT,ROC,DISTMAX,SINKD,RDSCALE,RAM,PHONY
            ICOUNT=0 $ RUN=RUNDATA(1)=AGDATA(1)=PHDATA(1)=0.
         10 W=SEGMENT=NPRINT=IPRINT=NODRAG=NOSTORE=METRIC=NODOEI=0
            FLOSSA=FLOSSF=0.
            DT=.05          $       RAM=1.
            REVTIME=ANGFRL=ALPHMIN=EDRAG=OSTORE =V=T=ALT=D=EL=WF1=PHI1=PHI2=
           *FN1=FN2=DLFG=DDFB       =VSTARR=EDATUM=EFAIL=DEI=DEILOSF=0.
            DDEILOF=RSF=PHI1AR=PHI1RR=PSINT=AGY=DROC=ROC=WDD=ROCH=PHONY=0.
           *F2=RS=DELTAE=PHI2AR=PHI2RR=ALPHARR=ALPHAAR=RUN=PS=DEITAB=DD=0.
            FLOTEMP=59. $ MU=.02 $ NEA=NEF=1.    $ G1=1. $ G2=2. $ DISTMAX=1.E5
            SINKD=ASINK=VSINK=DSINK=0.      $ RDSCALE=1.
            IF(ICOUNT.NE.0) GO TO 90
            IC=1
         20 READ 30,ICOPY
         30 FORMAT(2A1,A8,7A10)
         40 FORMAT(1X,2A1,A8,7A10)
            IF(EOF(9)) 60,50
         50 IC=IC+1
            WRITE(8,30) ICOPY
            PRINT 40,ICOPY
            IF(IC.NE.2.AND.ICOPY(2).EQ.1H$) GO TO 60
            GO TO 20
         60 DO 70 II=1,IC
         70 BACKSPACE 9
            READ DATA
            READ80,TITLE
            CALL TREAS(1,WD,XD,YD,ZD,FWXYZ)
         80 FORMAT(8A10)
            GO TO 110
         90 REWIND 8
            READ(8,DATA)
            READ 30,ICOPY
            IF(EOF(9)) 750,100
        100 PRINT 40,ICOPY
            BACKSPACE 9 $ READ DATA
        110 PH2I=PHI2
            ALPHAO=ALPHA
            PHI1I=PHI1
            PHI1=(PHI1+ANGFRL)*3.14159265/180.
            IF(SEGMENT.EQ.0) GO TO 120
            LEVER=LRA+MU*LR     $ OLEVER=LRA+LRF
        120 V=V+WDD
            IF(ICOUNT.EQ.0) PRINT 130,TITLE
```

```
  130 FORMAT(1H1,//,10X,*VSTOL TAKEOFF PERFORMANCE ESTIMATION*,/,10X,35(STD 0058
     (1H-),//,3A10,//)                                                STD 0059
      ICOUNT=0                                                        STD 0060
      TOGW=W                                                          STD 0061
      PRINT 140,W                                                     STD 0062
  140 FORMAT(10X,*TOGW =*,E20.5,//,10X,                               STD 0063
     **INITIAL GROUND RUN PARAMETERS*,/)                              STD 0064
      IF(IPRINT.NE.0) PRINT 150                                       STD 0065
  150 FORMAT(10X,120(1H$),/,10X,*W,V,L,DR,RF,EDRG,FN2*,/,10X,*PHI2,ALPHASTD 0066
     *,T,D2,BALANCE,PHI1,FN1*,/,10X,*CL,CD,CM,A/G,RHO,DELTAE*,/,10X,*ALPSTD 0067
     *HA,DALPHAD,IYYCG,XCG,YCG*,/,10X,*N,RF,DELTAE*,/,10X,*ROC,ACCY*,/,1STD 0068
     *0X,*ASINK,VSINK,DSINK*,/,10X,120(1H$))                          STD 0069
      IF(METRIC.EQ.0) ALT=ALT*.3048                                   STD 0070
      CALL ATMOS(ALT,A1,A2,A3,A4,A5,A6,A7,A8)                         STD 0071
      IF(METRIC.NE.0) GO TO 160                                       STD 0072
      ALT=ALT*3.280839895                                             STD 0073
      A6=A6*3.280839895                                               STD 0074
      A3=A2*2.3769E-3                                                 STD 0075
      AFLOTEM=FLOTEMP+459.7                                           STD 0076
      A6=A6*SQRT(AFLOTEM/518.7)                                       STD 0077
      A3=A3*518.7/AFLOTEM                                             STD 0078
      GO TO 170                                                       STD 0079
  160 AFLOTEM=FLOTEMP+273.16                                          STD 0080
      A6=A6*SQRT(AFLOTEM/288.16)                                      STD 0081
      A3=A3*288.16/AFLOTEM                                            STD 0082
  170 EDATUM=EDATUM*3.141592654/180.                                  STD 0083
      G=32.1741                                                       STD 0084
      IF(METRIC.NE.0) G=9.8                                           STD 0085
      GOGC=1. $ IF (METRIC.NE.0) GOGC=9.8                             STD 0086
      W=W*GOGC                                                        STD 0087
      J=-1 $ WRC=0. $ COUNT=(.25/DT)+1.0E-10 $ I=COUNT $ M=I+1        STD 0088
  180 MACH=V/A6                                                       STD 0089
      J=J+1 $ IF(J.GE.M) J=1                                          STD 0090
      TOUM=T     $           IF(TOUM.GT.REVTIME)  TOUM=REVTIME        STD 0091
      PHI2=PH2I                                                       STD 0092
      IF(SEGMENT.NE.0) GO TO 210                                      STD 0093
C******** FORE ENGINE NOZZLE VARIATION                                STD 0094
      IF(PHI1AR.EQ.0..OR.V.LT.RSF) GO TO 190                          STD 0095
      PHI1=PHI1+PHI1RR*DT                                             STD 0096
      IF(PHI1.GT.PHI1AR) PHI1=PHI1AR                                  STD 0097
      PHI1=(PHI1+ANGFRL)*3.141592654/180.                             STD 0098
C  FOR ANY FUTURE TABLE LOOK UPS USE PHI1                             STD 0099
C*********************************************                        STD 0100
  190 IF(V.LT.RS) GO TO 200                                           STD 0101
                 PHI2=PHI2+PHI2RR*DT                                  STD 0102
      IF(PHI2.GT.PHI2AR) PHI2=PHI2AR                                  STD 0103
  200 IF(V.LT.PS) GO TO 210                                           STD 0104
      ALPHA=ALPHA+ALPHARR*DT                                          STD 0105
      IF(ALPHA.GT.ALPHAAR) ALPHA=ALPHAAR                              STD 0106
  210 IF(E1.NE.0.) CALL TLOO5(011,MACH,O1,TOUM,0.,WF1)                STD 0107
      CALL TLOO5(11,MACH,O2,TOUM,0.,WF2)                              STD 0108
      WF1=WF1*NEF*GOGC                                                STD 0109
      WF2=WF2*NEA*GOGC                                                STD 0110
      IF(E1.NE.0.) CALL TLOO5(10,MACH,O1,TOUM,0.,FN1)                 STD 0111
      CALL TLOO5(10,MACH,O2,TOUM,0.,FN2)                              STD 0112
      CALL TLOO5(12,PHI1,1.,0.,0.,FLOSSF)                             STD 0113
      CALL TLOO5(12,PHI2,2.,0.,0.,FLOSSA)                             STD 0114
```

```
PROGRAM STD        74/74   OPT=1                    FTN 4.6+428         81/09/28. 10.18.43          PAGE     3

                FN1=(FN1-FLOSSF*FN1)*NEF                                              STD 0115
                FN2=(FN2-FLOSSA*FN2)*NEA                                              STD 0116
          C ********   INDUCED AERO                                                   STD 0117
                FGTOTAL=FN1+FN2                                                       STD 0118
                CALL TLOOS(15,PHI2,1.,0.,0.,DLFG)                                     STD 0119
                CALL TLOOS(15,PHI2,2.,0.,0.,DDFG)                                     STD 0120
                DLFG=DLFG*FGTOTAL $ DDFG=DDFG*FGTOTAL                                 STD 0121
          C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$                   STD 0122
                PH2I=PHI2                                                             STD 0123
                VSTARR1=V     $       IF(V.LT.VSTARR) VSTARR1=VSTARR                  STD 0124
                CALL TLOOS(14,ALPHA,DELTAE,PH2I,VSTARR1,CL)                           STD 0125
                CALL TLOOS(13,CL,DELTAE,PH2I,VSTARR1,CD)                              STD 0126
                IF(SEGMENT.NE.0) CALL TLOOS(17,CL,DELTAE,PH2I,VSTARR1,CM)             STD 0127
                PHI2=(PHI2+ANGFWL)*3.141592654/180.                                   STD 0128
                DR=.5*CD*V**2*A3*S                                                    STD 0129
                CALL TLOOS(33,MACH,D1,0.,0.,EDRAG)                                    STD 0130
                CALL TLOOS(33,MACH,D2,0.,0.,DSTORE)                                   STD 0131
                EDRAG=EDRAG*ROSCALE                                                   STD 0132
                IF(NOEDRAG.EQ.0) EDRAG=0                                              STD 0133
                IF(NOSTORE.EQ.0) DSTORE=0                                             STD 0134
                IF(DEI.NE.0..AND.NODDEI.NE.0.)CALL TLOOS(33,MACH,3.,0.,0.,DCEILOF)STD 0135
                DR=DR+EDRAG+DSTORE+DDEILOF+DDFG                                       STD 0136
                PITCH=(ALPHA-ALPHAD)*3.141592654/180.                                 STD 0137
                ANG1=PHI1+PITCH+EDATUM $ ANG2=PHI2+PITCH+EDATUM                       STD 0138
                L=.5*CL*V**2*A3*S                                                     STD 0139
                L=L+DLFG                                                              STD 0140
                FGV=FN1*SIN(ANG1)+FN2*SIN(ANG2)                                       STD 0141
                FGH=FN1*COS(ANG1)+FN2*COS(ANG2)                                       STD 0142
                BALANCE=W-L-FGV                                                       STD 0143
                DV=DT*G       *(FGH-DR-MU*BALANCE)/W                                  STD 0144
                V2=V+DV                                                               STD 0145
                IF(SEGMENT.EQ.0) GO TO 220                                            STD 0146
                MOMENT=.5*CM*V**2*A3*S*MAC                                            STD 0147
                RF=(LEVER*BALANCE-MOMENT)/DLEVER                                      STD 0148
                IF(PITCH.LE.0.) GO TO 220                                             STD 0149
                ANGMESS=LRA*COS(PITCH)+MU*LRA*SIN(PITCH)+LR*MU*COS(PITCH)-LR*SIN(PSTD 0150
               *ITCH)                                                                 STD 0151
                RF=(BALANCE*ANGMESS-MOMENT)/(LRF+ANGMESS*COS(PITCH))                  STD 0152
            220 D2=D+.5*DT*((V2-WDD)**2.-(V-WDD)**2.)/DV                              STD 0153
                IF(IPRINT.EQ.0.OR.J.NE.I) GO TO 250                                   STD 0154
                ACCX=(FGH-DR-MU*BALANCE)/W                                            STD 0155
                PRINT 230,W,V,L,DR,RF,EDRAG,FN2                                       STD 0156
                PRINT 230,PH2I ,ALPHA,T ,D2,BALANCE,PHI1,FN1                          STD 0157
                PRINT 240,CL,CD,CM,ACCX,        A3,DELTAE                             STD 0158
            230 FORMAT(5X,7(5X,E12.5))                                                STD 0159
            240 FORMAT(5X,6(5X,E12.5),/,130(1H?))                                     STD 0160
            250 IF(SEGMENT.EQ.0) GO TO 260                                            STD 0161
                IF(RF.LE.0.) GO TO 520                                                STD 0162
            260 IF(BALANCE.GT.0..OR.DEITAB.EQ.0.) GO TO 230                           STD 0163
                DD=1.         $     J=J-1                                             STD 0164
                DEITAB=0.                                                             STD 0165
                ROSCALE=RAM                                                           STD 0166
                IF(RAM.EQ.1.)D1=3.                                                    STD 0167
                IF(RAM.EQ.1.)D2=4.                                                    STD 0168
                IF(PHONY.EQ.0.) GO TO 180                                             STD 0169
                PHI1=PHI1AR $ PH2I=PHI2AR $ ALPHA=ALPHAAR                             STD 0170
                PRINT 270                                                             STD 0171
```

```
270 FORMAT(10X,*INSTANTANEOUS ROTATION AT DEI*)                     STD 0172
    PHONY=-.13 $ GO TO 180                                          STD 0173
280 IF(BALANCE.GT.0..OR.EFAILA.EQ.0..OR.DEI.NE.0.) GO TO 290        STD 0174
    DEI=1. $ NEA=NEA*.5 $ NEF=NEF*DECLOSE $ GO TO 180               STD 0175
290 D=D2 $ V=V2 $ T=T+DT $ W=W-((WF1+WF2)*DT/3600.)                 STD 0176
    IF(PHONY.EQ.-.13) GO TO 350                                     STD 0177
    IF(D.LT.DISTMAX) GO TO 310                                      STD 0178
    PRINT 300                                                       STD 0179
300 FORMAT(10X,*CASE TERMINATED (DISTMAX EXCEEDED)*)                STD 0180
    GO TO 730                                                       STD 0181
310 IF(ICOUNT.GT.1000) GO TO 410                                    STD 0182
    IF(BALANCE.GE.0..OR.ROC.EQ.0.) GO TO 330                        STD 0183
    MU=0.$ AGYL=AGY $ DROCL=DROC                                    STD 0184
    AGY=(-BALANCE)*G/W                                              STD 0195
    AGYA=(AGY+AGYL)*.5                                              STD 0186
    DROC=DROCL+AGYA*DT                                              STD 0187
    AROC=(DROC+DROCL)*.5                                            STD 0188
    ROCH=ROCH+AROC*DT                                               STD 0189
    IF(IPRINT.NE.0.AND.J.EQ.I) PRINT 320,DROC,AGYA                  STD 0190
320 FORMAT(1X,*CLIMB DATA*,2(10X,E20.5))                            STD 0191
    IF(DROC.GE.ROC) GO TO 430                                       STD 0192
    GO TO 350                                                       STD 0193
330 IF(SINKD.GT.0..AND.SINKD.LE.0) GO TO 900                        STD 0194
340 IF(BALANCE.LE.0.) GO TO 430                                     STD 0195
350 ICOUNT=ICOUNT+1                                                 STD 0196
    GO TO 180                                                       STD 0197
360 IF(METRIC.NE.0) GO TO 380                                       STD 0198
    PRINT 370,D,V,T,W                                               STD 0199
370 FORMAT(10X,*DISTANCE = *,E20.5,* FT.*,/,10X,*VELOCITY = *,E20.5,* STD 0200
   (FT./SEC.*,/,10X,*TIME =*,5X,E20.5,* SEC.*,/,10X,*END WEIGHT FOR THSTD 0201
   (IS PHASE IS *,E20.5,* POUNDS MASS*)                             STD 0202
    GO TO 400                                                       STD 0203
380 PRINT 390,D,V,T,W                                               STD 0204
390 FORMAT(10X,*DISTANCE = *,E20.5,*  M.*,/,10X,*VELOCITY = *,E20.5,* STD 0205
   ( M./SEC.*,/,10X,*TIME =*,5X,E20.5,* SEC.*,/,10X,*END WEIGHT FOR THSTD 0206
   (IS PHASE IS *,E20.5,* KILOGRAMS  *)                             STD 0207
400 IF(SEGMENT.EQ.0) GO TO 630                                      STD 0208
    GO TO (500,610), SEGMENT                                        STD 0209
410 PRINT 420                                                       STD 0210
420 FORMAT(10X,*TOO MUCH TIME*)                                     STD 0211
    GO TO 360                                                       STD 0212
430 PRINT 440                                                       STD 0213
440 FORMAT(10X,*VEHICLE IS AIRBORNE*)                               STD 0214
    ISINK=10H FT.                                                   STD 0215
    IF(METRIC.NE.0) ISINK=10H M.                                    STD 0216
    IF(SINKD.GT.0.) PRINT 450,DSINK,ISINK                           STD 0217
450 FORMAT(10X,*MAXIMUM SINK DISTANCE IS*,E20.5,A10)                STD 0218
    IF(DEI.NE.0..OR.DD.NE.0.) PRINT 460                             STD 0219
460 FORMAT (10X,*DEI AT PSEUDO LIFTOFF*)                            STD 0220
    IF(ROC.EQ.0.) GO TO 490                                         STD 0221
    IROC=10H FT./SEC.                                               STD 0222
    IF(METRIC.NE.0) IROC=10H M./SEC.                                STD 0223
    PRINT 470,ROC,IROC                                              STD 0224
470 FORMAT(10X,*VEHICLE HAS A R/C OF *,E20.5,A10)                   STD 0225
    IROC=10H FT.                                                    STD 0226
    IF(METRIC.NE.0) IROC=10H M.                                     STD 0227
    PRINT 480,ROCH,IROC                                             STD 0228
```

```
   480 FORMAT(10X,*VERTICAL HEIGHT IS *,E20.5,A10)                   STO 0229
   490 GO TO 360                                                     STO 0230
   500 PRINT 510                                                     STO 0231
   510 FORMAT(10X,*AUTO ROTATION IS OCCURING*)                       STO 0232
       GO TO 630                                                     STO 0233
C       ALPHA CORRECTION     AUTO ROTATION$$$$$$$$$$$$$$$$$$$$$$$$$$$ STO 0234
   520 IF(SEGMENT.GT.1.AND.ALPHA.GE.ALPHMIN) GO TO 550               STO 0235
   530 CALL TLOO5(30,W,0.,0.,0.,IYYCG)                               STO 0236
       CALL TLOO5(31,W,1.,0.,0.,XCG)                                 STO 0237
       CALL TLOO5(31,W,2.,0.,0.,YCG)                                 STO 0238
       DCGS=XCG**2+YCG**2 $ LRA=XCG $ LRF=DLEVER-XCG                 STO 0239
       IYY=IYYCG+W/G*DCGS $ IF(WRC.NE.0.)GO TO 531 $ WR=LR-YCG $ WRC=1. STO 0240
   531 LR=YCG+WR $ DALPHAR=DT**2*(-RF)*DLEVER/IYY                    STO 0241
       DALPHAD=DALPHAR*180./3.141592654                              STO 0242
       ALPHA=ALPHA+DALPHAD                                           STO 0243
       IF(SEGMENT.GT.1.AND.ALPHA.GE.ALPHMIN.AND.ITRIP.EQ.0) GO TO 550 STO 0244
       ITRIP=0                                                       STO 0245
       IF(IPRINT.NE.0.AND.J.EQ.I) PRINT 540,ALPHA,DALPHAD,IYYCG,XCG,YCG STO 0246
   540 FORMAT(10X,5(5X,E12.5),/,1X,3(15H AUTO ROTATION ))            STO 0247
       GO TO 260                                                     STO 0248
C$$$$$$$$$$ $$$$$$$ TRY TO COUNTER ROTATION BY ELEVATOR               STO 0249
   550 DDELTAE=1. $ ITRIP=0                                          STO 0250
   560 CALL SNEST(N,RF,DELTAE,DDELTAE,010.,ABDS,DYL,ABDYL,XCHECK,XLOC) STO 0251
       IF(IPRINT.NE.0.AND.J.EQ.I) PRINT 570,N,RF,DELTAE              STO 0252
   570 FORMAT(10X,25(1H*),5X,I10,2(10X,E12.5))                       STO 0253
       IF(DELTAE.LT.DELMAX) GO TO 580                                STO 0254
       DELTAE=DELMAX $ N=0 $ ITRIP=1                                 STO 0255
   580 CALL TLOO5(14,ALPHA,DELTAE,PH2I,VSTARR1,CL)                   STO 0256
       CALL TLOO5(13,CL,DELTAE,PH2I,VSTARR1,CD)                      STO 0257
       CALL TLOO5(17,CL,DELTAE,PH2I,VSTARR1,CM)                      STO 0258
       DR=.5*CD*V**2*A3*S                                            STO 0259
       DR=DR+EDRAG+DSTORE+DOEILOF+DOFG                               STO 0260
       L=.5*CL*V**2*A3*S                                             STO 0261
       L=L+DLFG                                                      STO 0262
       BALANCE=W-L-FGV                                               STO 0263
       DV=DT*G      *(FGH-DR-MU*BALANCE)/W                           STO 0264
       V2=V+DV                                                       STO 0265
       MOMENT=.5*CM*V**2*A3*S*MAC                                    STO 0266
       RF=(LEVER*BALANCE-MOMENT)/DLEVER                              STO 0267
       IF(PITCH.LE.0.) GO TO 590                                     STO 0268
       ANGRESS=LRA*COS(PITCH)+MU*LRA*SIN(PITCH)+LR*MU*COS(PITCH)-LR*SIN(PSTO 0269
      *ITCH)                                                         STO 0270
       RF=(BALANCE*ANGRESS-MOMENT)/(LRF+ANGRESS*COS(PITCH))          STO 0271
   590 D2=D+.5*DT*((V2-WDD)**2.-(V-WDD)**2.)/DV                      STO 0272
       IF(ITRIP) 530,500                                             STO 0273
   600 IF(N) 560,260                                                 STO 0274
   610 PRINT 620                                                     STO 0275
   620 FORMAT(10X,*ELEVATOR CORRECTION ATTEMPTED*)                   STO 0276
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++STO 0277
C                         LIFTOFF CHARATERISTICS                      STO 0278
C++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++STO 0279
   630 PRINT 640                                                     STO 0280
   640 FORMAT(//,10X,*LIFTOFF CHARACTERISTICS*,/)                    STO 0281
       AGX=(FGH-DR-MU*BALANCE)/W                                     STO 0282
       IF(PHONY.NE.-.13) GO TO 670                                   STO 0283
       PHRC=V*60.*((L+FGV-W)/FGH)                                    STO 0284
       IROC=104 FT/MIN                                               STO 0285
```

```
      IF(METRIC.EQ.0) GO TO 650                                         STD 0286
      IROC=10H M/SEC                                                    STD 0287
      PHRC=PHRC/60.                                                     STD 0288
  650 PRINT 660,PHRC,IROC                                               STD 0289
  660 FORMAT(10X,*INSTANTANEOUS R/C IS *,E20.5,A10)                     STD 0290
  670 IF(METRIC.NE.0) GO TO 590                                         STD 0291
      PRINT 680,AGX,L,DR,FGV,FGH                                        STD 0292
  680 FORMAT(10X,*HORIZONTAL ACCELERATION =*,E20.5,* G(S*,/,10X,        STD 0293
     **LIFT =*,19X,E20.5,* POUNDS FORCE*,/,10X,*DRAG =*,19X,E20.5,      STD 0294
     ** POUNDS FORCE*,/,10X,*VERTICAL GROSS THRUST =*,2X,E20.5,         STD 0295
     ** POUNDS FORCE*,/,10X,*HORIZONTAL GROSS THRUST =*,E20.5,          STD 0296
     ** POUNDS FORCE*)                                                  STD 0297
       GO TO 710                                                        STD 0298
  590 PRINT 700,AGX,L,DR,FGV,FGH                                        STD 0299
  700 FORMAT(10X,*HORIZONTAL ACCELERATION =*,E20.5,* G(S*,/,10X,        STD 0300
     **LIFT =*,19X,E20.5,* NEWTONS      *,/,10X,*DRAG =*,19X,E20.5,     STD 0301
     ** NEWTONS      *,/,10X,*VERTICAL GROSS THRUST =*,2X,E20.5,        STD 0302
     ** NEWTONS      *,/,10X,*HORIZONTAL GROSS THRUST =*,E20.5,         STD 0303
     ** NEWTONS      *)                                                 STD 0304
  710 IF(RUN.LE.0.) GO TO 730                                           STD 0305
      CALL UPDATE(AGDATA,D,AGX)                                         STD 0306
      IF(PHONY.EQ.-.13) CALL UPDATE(PHDATA,D,PHRC)                      STD 0307
      IF(PSINT.LE.0.) GO TO 720                                         STD 0308
      CALL UPDATE(RUNDATA,D,PS)             $ GO TO 730                 STD 0309
  720 CALL UPDATE(RUNDATA,D,TOGW)                                       STD 0310
  730 PRINT 740                                                         STD 0311
  740 FORMAT(/,1X,22(6H NEXT ),/)                                       STD 0312
      GO TO 10                                                          STD 0313
  750 IF(RUN.LE.0.) STOP                                                STD 0314
      IF(PHDATA(1).EQ.0.) GO TO 770                                     STD 0315
      PHROC=SPLN01(1,PHDATA,RUN)                                        STD 0316
      PRINT 760,PHROC,IROC                                              STD 0317
  760 FORMAT(10X,*INSTANTANEOUS INTERPOLATED R/C IS *,E20.5,A10)        STD 0318
  770 AG=SPLN01(1,AGDATA,RUN)                                           STD 0319
      WRUN=SPLN01(1,RUNDATA,RUN)                                        STD 0320
      K(1)=6H FEET  $ K(2)=6H TOGW  $ K(3)=10H POUNDS MA  $ K(4)=2HSS   STD 0321
      IF(METRIC.EQ.0.AND.PSINT.EQ.0.) GO TO 790                         STD 0322
      K(1)=6H   M.  $ K(3)=10H KILOGRAMS  $ K(4)=2H                     STD 0323
      IF(PSINT.EQ.0.) GO TO 780                                         STD 0324
      K(2)=6H   PS    $ K(3)=10H M./SEC.                                STD 0325
      IF(METRIC.NE.0) GO TO 780                                         STD 0326
      K(1)=6H FEET  $ K(3)=10H FT./SEC.                                 STD 0327
  780 PRINT 790,RUN,K(1),K(2),WRUN,K(3),K(4),AG                         STD 0328
  790 FORMAT(10X,*BY CUBIC SPLINE INTERPOLATION OF THE ABOVE DATA POINTSSTD 0329
     **,/,25X,*FOR A RUN OF *,E20.5,A6,*THE*,A5,*IS*,,E20.5,A10,A3,/,60XSTD 0330
     *,*AND THE HORIZONTAL ACCELERATION IS *,E20.5,* G(S*)              STD 0331
      STOP                                                              STD 0332
C????????????????????        SINK OFF THE BOW CALCULATIONS              STD 0333
  800 IF(MJ.EQ.0) GO TO 810                                             STD 0334
      IF(PS.GT.0.) PS=V $ IF(RS.GT.0.) RS=V $ IF(RSF.GT.0.) RSF=V $ MU=0STD 0335
  810 ASINKL=ASINK $ VSINKL=VSINK $ DSINKL=DSINK                        STD 0336
      ASINK=BALANCE*G/W                                                 STD 0337
      ASINKA=(ASINK+ASINKL)*.5                                          STD 0338
      VSINK= VSINKL+ASINKA*DT                                           STD 0339
      VSINKA=(VSINK+VSINKL)*.5                                          STD 0340
      DSINK=DSINKL+VSINKA*DT                                            STD 0341
      IF(IPRINT.NE.0.AND.J.EQ.I) PRINT 920,ASINK,VSINK,DSINK            STD 0342
```

```
PROGRAM STD        74/74   OPT=1                         FTN 4.6+428        81/09/29. 10.19.43       PAGE     7

       820 FORMAT(1X,*SINK DATA*,3(10X,E20.5))                                   STD 0343
           GO TO 340                                                             STD 0344
      C**************************************************************************STD 0345
           END                                                                   STD 0346
```

```
SUBROUTINE TAKE5      74/74   OPT=1                     FTN 4.5+42C          81/09/23. 10.13.43        PAGE     1

          CTAKE5:                                                                  TAK50001
                SUBROUTINE TAKE5(II,J,K,Y,Z,F,XYZ)                                 TAK50002
          C***  MODIFIED GTREAD TO INCLUDE FOUR INDEPENDENT VARIABLES- DSK 3/77     TAK50003
                COMMON /TSIZE/NMAX,TDATA(3300)                                     TAK50004
                COMMON/PRINT/NPRINT                                                TAK50005
                DIMENSION A(100),IDD(5),SIRIUS(30)                                 TAK50006
                DATA NMAX,NTBL/3300,0/                                             TAK50007
                DATA NPRINT/0/                                                     TAK50008
                ENTRY TREAS                                                        TAK50009
                MRIGEL=0                                                           TAK50010
                ITPRT    = NPRINT                                                  TAK50011
                IF(NTBL.NE.0) GO TO 20                                             TAK50012
                LOC=61                                                             TAK50013
                DO 10 M=1,NMAX                                                     TAK50014
             10 TDATA(M)=0.                                                        TAK50015
                ILAST=1                                                            TAK50016
             20 READ 30, IP,ITABNO                                                 TAK50017
             30 FORMAT (I1,I4,75X                                                  TAK50018
               1                               )                                   TAK50019
                IF(ITABNO.EQ.0) GO TO 110                                          TAK50020
                NTBL     = NTBL+1                                                  TAK50021
                ENIF=TDATA(NTBL)=ITABNO                                            TAK50022
                TDATA(NTBL+30)=LOC                                                 TAK50023
             40 IF(NTBL.LE.1) GO TO 210                                            TAK50024
                DO 50 LYRA=2,NTBL                                                  TAK50025
                IF(ENIF.EQ.TDATA(LYRA-1)) GO TO 60                                 TAK50026
             50 CONTINUE                                                           TAK50027
                GO TO 210                                                          TAK50028
             60 IARIES=MERAK=TDATA(LYRA+29)                                        TAK50029
                MRIGEL=MRIGEL+1                                                    TAK50030
                SIRIUS(MRIGEL)=ENIF                                                TAK50031
                LIBRAM=100000                                                      TAK50032
                IZAR=LOC-1                                                         TAK50033
                DO 70 IRK=1,NTBL                                                   TAK50034
                LIBRA=TDATA(IRK+30)                                                TAK50035
                IF((LIBRA.GT.MERAK).AND.(LIBRA.LT.LIBRAM)) LIBRAM=LIBRA            TAK50036
             70 CONTINUE                                                           TAK50037
                LEO=LIBRAM-MERAK                                                   TAK50038
                DO 80 NEKKAR=LIBRAM,IZAR                                           TAK50039
                TDATA(MERAK)=TDATA(NEKKAR)                                         TAK50040
             80 MERAK=MERAK+1                                                      TAK50041
                DO 90 NEKKAR=1,NTBL                                                TAK50042
                LLOC=TDATA(NEKKAR+30)                                              TAK50043
                IF(LLOC.GT.IARIES) TDATA(NEKKAR+30)=LLOC-LEO                       TAK50044
             90 CONTINUE                                                           TAK50045
                TDATA(LYRA+29)=LOC-LEO                                             TAK50046
                DO 100 IBALT=MERAK,LOC                                             TAK50047
            100 TDATA(IBALT)=0.                                                    TAK50048
                LOC=MERAK                                                          TAK50049
                NTBL=NTBL-1                                                        TAK50050
                GO TO 210                                                          TAK50051
            110 NMEM=NMAX-LOC                                                      TAK50052
                IF(NTBL.LE.1) GO TO 140                                            TAK50053
          C*** RE ORDER TABLE REFERENCE NUMBERS                                    TAK50054
            120 NDONE=1                                                            TAK50055
                DO 130 L=2,NTBL                                                    TAK50056
                K1=L-1                                                             TAK50057
```

```
SUBROUTINE TAKED      74/74    OPT=1                          FTN 4.6+428          81/03/26. 10.18.43          PAGE     2

        K2=L                                                                  TAK50058
        IF(TDATA(K1).LT.TDATA(K2)) GO TO 130                                  TAK50059
        XSAVE=TDATA(K1)                                                       TAK50060
        TDATA(K1)=TDATA(K2)                                                   TAK50061
        TDATA(K2)=XSAVE                                                       TAK50062
        XSAVE=TDATA(K1+30)                                                    TAK50063
        TDATA(K1+30)=TDATA(K2+30)                                             TAK50064
        TDATA(K2+30)=XSAVE                                                    TAK50065
        NDONE=6                                                               TAK50066
    130 CONTINUE                                                              TAK50067
        IF(NDONE.EQ.0) GO TO 120                                              TAK50068
    140 IF(II.LE.0) GO TO 470                                                 TAK50069
        IF(NPRINT.EQ.0) GO TO 160                                             TAK50070
        PRINT 150                                                             TAK50071
    150 FORMAT(1H1)                                                           TAK50072
    160 PRINT 170,NTBL                                                        TAK50073
    170 FORMAT (    35X,24HTABLE DATA INPUT SUMMARY,1X,I2,1X,6HTABLES,//,25TAK50074
       1X,43HTABLE NUMBER  REFERENCE NUMBER   ARRAY LOCATION )                TAK50075
        PRINT 180,((N,TDATA(N),TDATA(N+30)),N=1,NTBL)                         TAK50076
    180 FORMAT (32X,I2,12X,F5.0,15X,F5.0)                                     TAK50077
        PRINT 190, NMAX,NREM                                                  TAK50078
    190 FORMAT (//,35X,24HDATA STORAGE ALLOCATION ,I4,/35X,24HDATA STORAGETAK50079
       1 NOT USED   ,I4,/)                                                    TAK50080
        IF(NRIGEL.GT.0) PRINT 200,(SIRIUS(IO),IO=1,NRIGEL)                    TAK50081
    200 FORMAT(10X,*THE FOLLOWING TABLES HAVE BEEN REPLACED*,5(5X,F10.0,*,TAK50082
       2*),/,(8(5X,F10.0,*,*)))                                               TAK50083
C       PRINT 1111,((M,TDATA(M)),M=1, LOC)                                    TAK50084
C1111 FORMAT(5(1X,I4,1X,E12.5))                                               TAK50085
        RETURN                                                                TAK50086
    210 IP=1                                                                  TAK50087
        IF(NPRINT.EQ.0) IP=0                                                  TAK50088
        PRINT 30, IP,ITABNO                                                   TAK50089
        IF(NPRINT.NE.0) PRINT 220                                             TAK50090
    220 FORMAT(///////)                                                       TAK50091
        ICC     = 0                                                           TAK50092
        L=LOC                                                                 TAK50093
        ID=44                                                                 TAK50094
        DO 230 ICL=1,5                                                        TAK50095
    230 IDD(ICL)=4H                                                           TAK50096
    240 IDL=ID                                                                TAK50097
        READ 250,ID,N,(A(I),I=1,N)                                            TAK50098
    250 FORMAT(A4,I3,3X,(T11,7F10.0))                                         TAK50099
        IF(ID.EQ.4HEOT ) GO TO 20                                             TAK50100
        ICC=ICC+1                                                             TAK50101
        IF(ICC.LE.5) IDD(ICC)=ID                                              TAK50102
        IF(ID.EQ.IDL)     GO TO 350                                           TAK50103
        IF(ID.NE.IDD(5))     GO TO 260                                        TAK50104
        IF(IDL.EQ.IDD(3))     GO TO 350                                       TAK50105
        LOC=LE                                                                TAK50106
        L=LX                                                                  TAK50107
        GO TO 280                                                             TAK50108
    260 TDATA(LOC)=N                                                          TAK50109
        L=LOC                                                                 TAK50110
        IF(ID.EQ.IDD(2)) GO TO 390                                            TAK50111
    270 IF(ID.EQ.IDD(4)) LX=L                                                 TAK50112
        IF(ID.NE.IDD(3))           GO TO 330                                  TAK50113
        LY=L                                                                  TAK50114
```

```
SUBROUTINE TAKES      74/74    OPT=1                         FTN 4.6+428        81/09/28. 10.13.43        PAGE     3

                  LZ=LZ+1                                                          TAK50115
                  GO TO 330                                                        TAK50116
              280 IF(IP.EQ.0) GO TO 330                                            TAK50117
                  LY=LY+1                                                          TAK50118
                  IF(NPRINT.NE.0) PRINT 290,IDD(1),TDATA(LW)                       TAK50119
              290 FORMAT(1X,A4,1H=,E13.5)                                          TAK50120
                  IF(NPRINT.NE.0) PRINT 300,IDD(2),TDATA(LZ),IDD(3),TDATA(LY)      TAK50121
              300 FORMAT(1X,A4,1H=,E13.5,1X,A4,1H=,E13.5)                          TAK50122
                  JF=0                                                             TAK50123
                  M=N                                                              TAK50124
                  LF=LK                                                            TAK50125
              310 NP=M                                                             TAK50126
                  IF(M.GT.8)NP=8                                                   TAK50127
                  M=M-NP                                                           TAK50128
                  LE=LF+NP                                                         TAK50129
                  LF=LF+1                                                          TAK50130
                  IF(NPRINT.NE.0) PRINT 320,IDD(4),(TDATA(I),I=LF,LE)              TAK50131
              320 FORMAT(20X,A4,1X,8E13.5)                                         TAK50132
                  LF=LE                                                            TAK50133
                  JE=JF+NP                                                         TAK50134
                  JF=JF+1                                                          TAK50135
                  IF(NPRINT.NE.0) PRINT 320,IDD(5),(A(I),I=JF,JE)                  TAK50136
                  JF=JE                                                            TAK50137
                  IF(M.GT.0) GO TO 310                                             TAK50138
              330 DO 340 I=1,N                                                     TAK50139
                  LOC=LOC+1                                                        TAK50140
              340 TDATA(LOC)=A(I)                                                  TAK50141
                  LE=LOC                                                           TAK50142
                  IF(IO.EQ.IDD(5)) TDATA(LOC+2)=1.                                 TAK50143
                  LOC=L+3*N+3                                                      TAK50144
                  IF(LOC.GT.NMAX)370,240                                           TAK50145
              350 TDATA(LOC)=TDATA(LX)                                             TAK50146
                  L=LOC                                                            TAK50147
                  DO 360 I=1,N                                                     TAK50148
                  LOC=LOC+1                                                        TAK50149
              360 TDATA(LOC)=TDATA(LX+I)                                           TAK50150
                  GO TO 280                                                        TAK50151
              370 PRINT 380, ITABNO                                                TAK50152
              380 FORMAT (1X,32H**********TABLE OVER FLOW ,TABLE ,I5,11H NOT LOADED) TAK50153
                  GO TO 40                                                         TAK50154
              390 LZ=LOC $ LW=LW+1 $ GO TO 270                                     TAK50155
                  ENTRY TLOOS                                                      TAK50156
            C*****************************************************************     TAK50157
                  IF(II.LE.0) RETURN                                               TAK50158
            C*** SEARCH FOR CORRECT TABLE                                          TAK50159
                  NH=NTBL+1                                                        TAK50160
                  NL=1                                                             TAK50161
                  K=ILAST                                                          TAK50162
                  XTAB=II                                                          TAK50163
                  KT=0                                                             TAK50164
              400 KT=KT+1                                                          TAK50165
                  IF(KT.GT.6) GO TO 470                                            TAK50166
                  IF((XTAB-TDATA(K)).EQ.0.) GO TO 430                              TAK50167
                  IF((XTAB-TDATA(K)).LT.0.) GO TO 410                              TAK50168
                  NL      = K                                                      TAK50169
                  GO TO 420                                                        TAK50170
              410 NH=K                                                             TAK50171
```

```
SUBROUTINE TAKES      74/74    OPT=1                    FTN 4.5+420       81/04/28. 10.18.43        PAGE      4

          420 K=(NH-NL)/2+NL                                              TAK50172
              GO TO 400                                                   TAK50173
          430 ILAST=K                                                     TAK50174
              IZ=TDATA(K+30)                                              TAK50175
              INZ=TDATA(IZ)                                               TAK50176
              IY=3*INZ+IZ+3                                               TAK50177
              IX=IY                                                       TAK50178
              IZ1=IZ+INZ+1                                                TAK50179
              IZ2=IZ+2*INZ                                                TAK50180
              DO 450 IZS=IZ1,IZ2                                          TAK50181
              INY=TDATA(IY)                                               TAK50182
              IX=3*INY+IX+3                                               TAK50183
              IW=IX                                                       TAK50184
              IY1=IY+INY+1                                                TAK50185
              IY2=IY+2*INY                                                TAK50186
              DO 450 IYS=IY1,IY2                                          TAK50187
              INX=TDATA(IX)                                               TAK50188
              IW=3*INX+IW+3                                               TAK50189
              IX1=IX+INX+1                                                TAK50190
              IX2=IX+2*INX                                                TAK50191
              DO 440 IXS=IX1,IX2                                          TAK50192
              TDATA(IXS)=SPLN01(IW,TDATA,W)                               TAK50193
          440 IW=IW+3*TDATA(IW)+3                                         TAK50194
              TDATA(IYS)=SPLN01(IX,TDATA,X)                               TAK50195
          450 IX=IW                                                       TAK50196
              TDATA(IZS)=SPLN01(IY,TDATA,Y)                               TAK50197
          460 IY=IX                                                       TAK50198
              FWXYZ=SPLN01(IZ,TDATA,Z)                                    TAK50199
          470 RETURN                                                      TAK50200
              END                                                         TAK50201
```

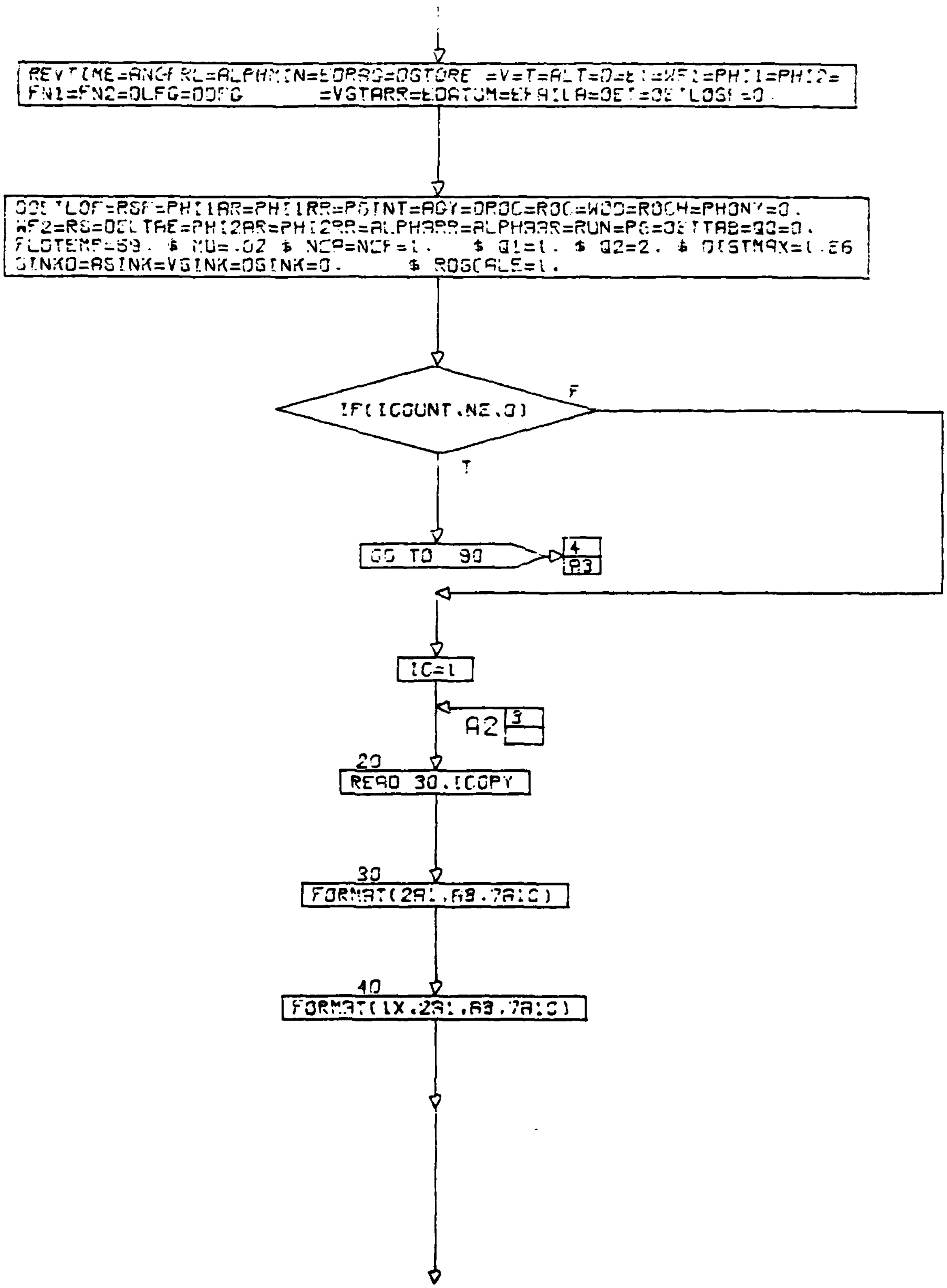
APPENDIX D

Program Logic

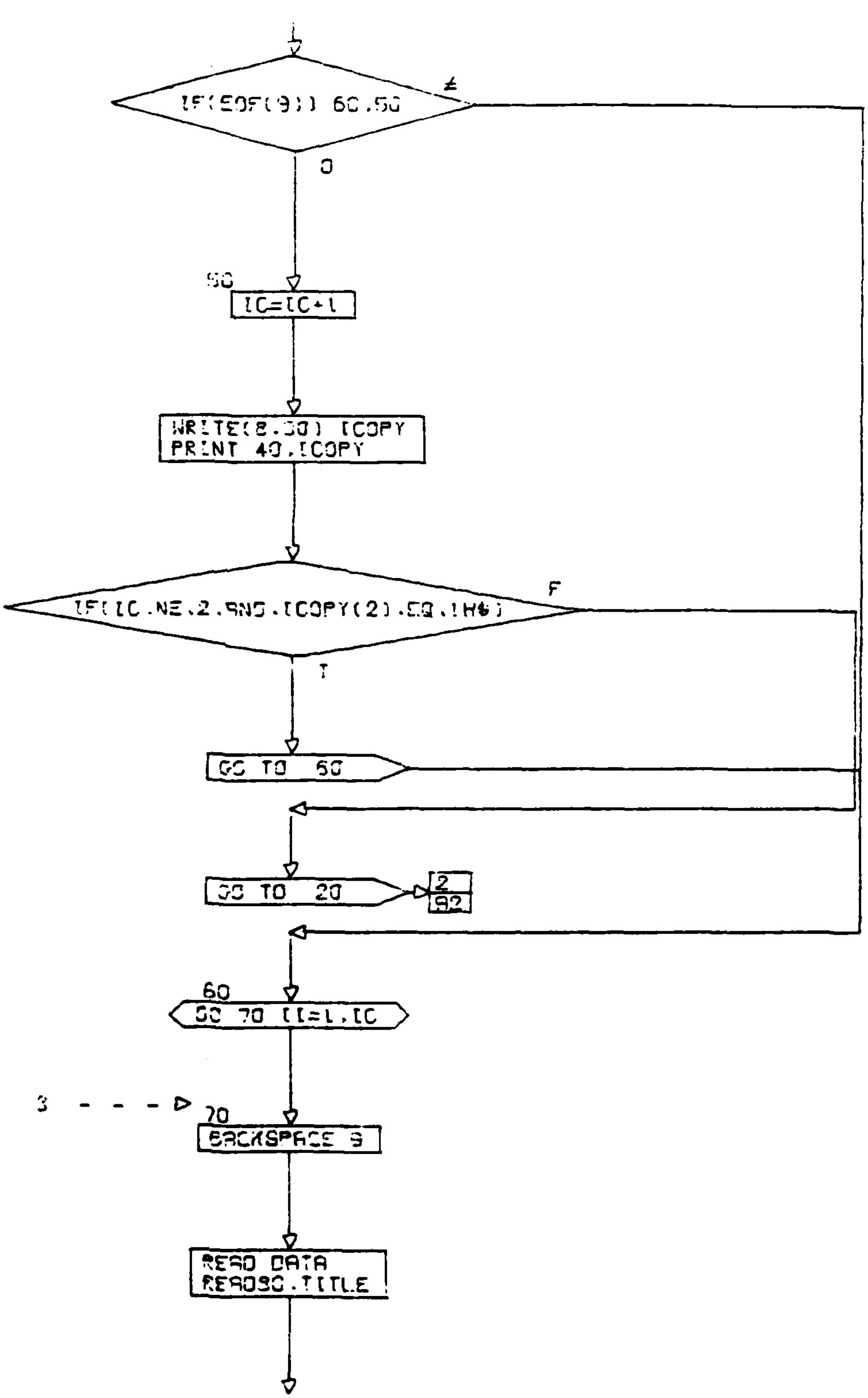
Flow charts for the main STO program and the subroutine TAKE5 are presented.

```
CST9

PROGRAM ST9 (INPUT,OUTPUT,TAPE8,TAPE9=INPUT)

C*******************************************************************
C   CREATED BY KOBUS ON APRIL 1977.
C*******************************************************************

DIMENSION TITLE(8),ICOPY(10),RUNDATA(33),X(4),AGDATA(33),
 PHDATA(33)

COMMON/PRINT/NPRINT
REAL MACH,NZF,NZA,MU,L,LR,LRA,LRF,MAC,MOMENT,LEVER,IYY,IYYCG
INTEGER SEGMENT

NAMELIST/DATA/ALPHA,S,DELTAE,NZF,NZA,MU,W,LR,LRA,LRF,REVTIME,DT,
IPRINT,NPRINT,VSTARR,MAC,SEGMENT,IYYCG,XCG,YCG,E1,DELMAX,PHI1,
ANGFRL,FLOTEMP,ALPHMIN,PHI2,PHI2AR,PHI2RR,RS,ALPHARR,ALPHAAR,SLT,
RUN,CDATUM,NOCORAG,NOSTORE,EFAILA,DEILOST,NOCOCI,METRIC,FS,WOD,
DEITAS,RSF,PHI1RR,PHI1AR,PSINT,ROC,DISTMAX,SINKD,ROSCALE,RAN,PHONY

ICOUNT=0 $ RUN=RUNDATA(1)=AGDATA(1)=PHDATA(1)=0.

A1 35

10
N=SEGMENT=NPRINT=IPRINT=NOCORAG=NOSTORE=METRIC=NOCOCI=0

FLOSSA=FLOSSF=0.
DT=.05          $        RAN=1.
```

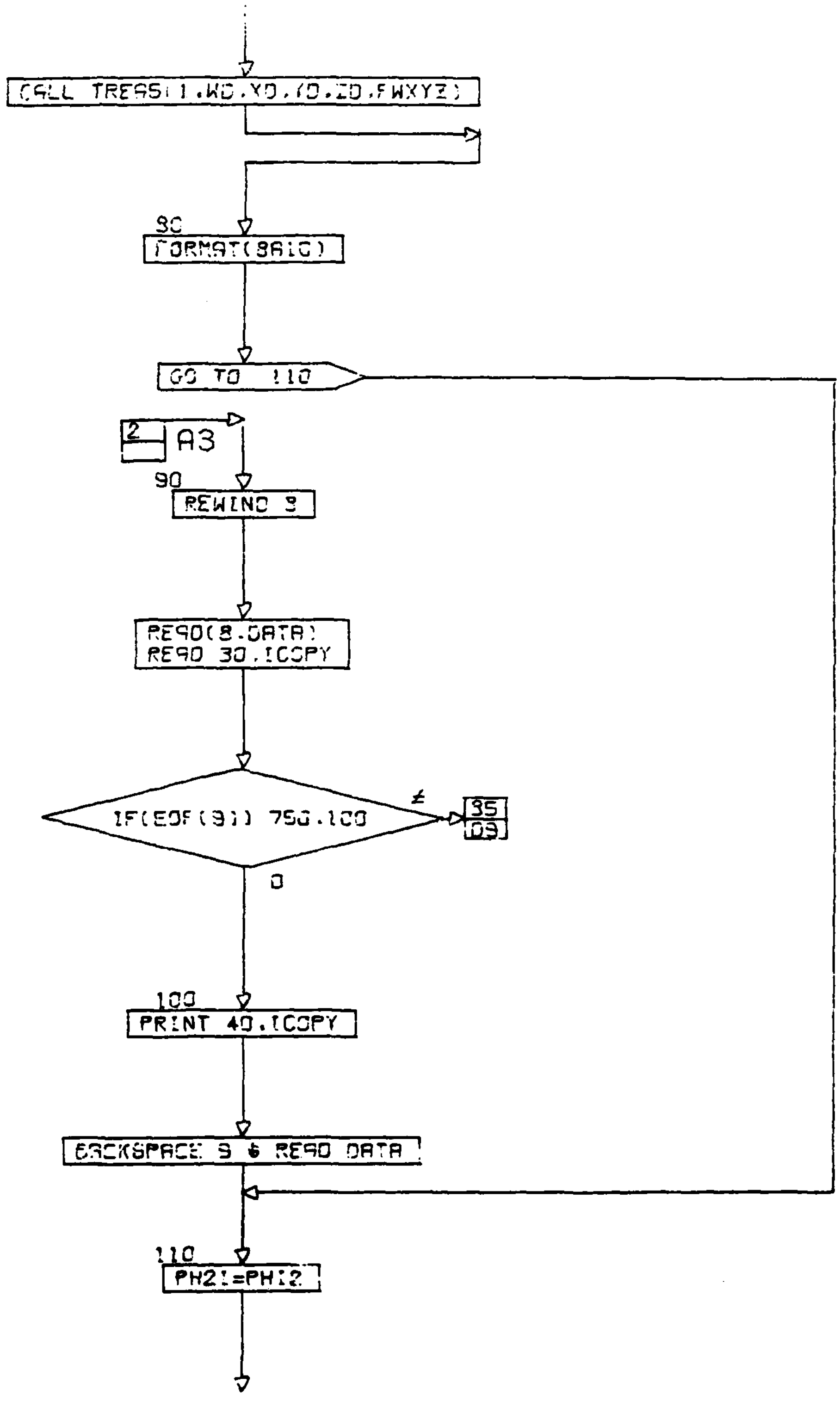
CONT. ON PG 2


REVTIME=ANGFRL=ALPHMIN=EDRAG=DSTORE =V=T=ALT=D=E1=WF1=PHI1=PHI2=
FN1=FN2=DLFG=DDFG =VGTARR=EDATUM=EFAILA=DET=DELOSS=0.
DELTLOF=RSF=PHI1AR=PHI1RR=PSINT=AGY=DROC=ROC=WDD=ROCH=PHONY=0.
WF2=RS=DELTAE=PHI2AR=PHI2RR=ALPHARR=ALPHAAR=RUN=PG=DETTAB=Q0=0.
FLOTEMP=59. $ MU=.02 $ NCP=NCF=1. $ Q1=1. $ Q2=2. $ DISTMAX=1.E6
SINKO=ASINK=VSINK=DSINK=0. $ ROSCALE=1.
IF(ICOUNT.NE.0)
F
T
GO TO 90
4
A3
IC=1
A2
3
20
READ 30,ICOPY
30
FORMAT(2A1,A9,7A10)
40
FORMAT(1X,2A1,A9,7A10)

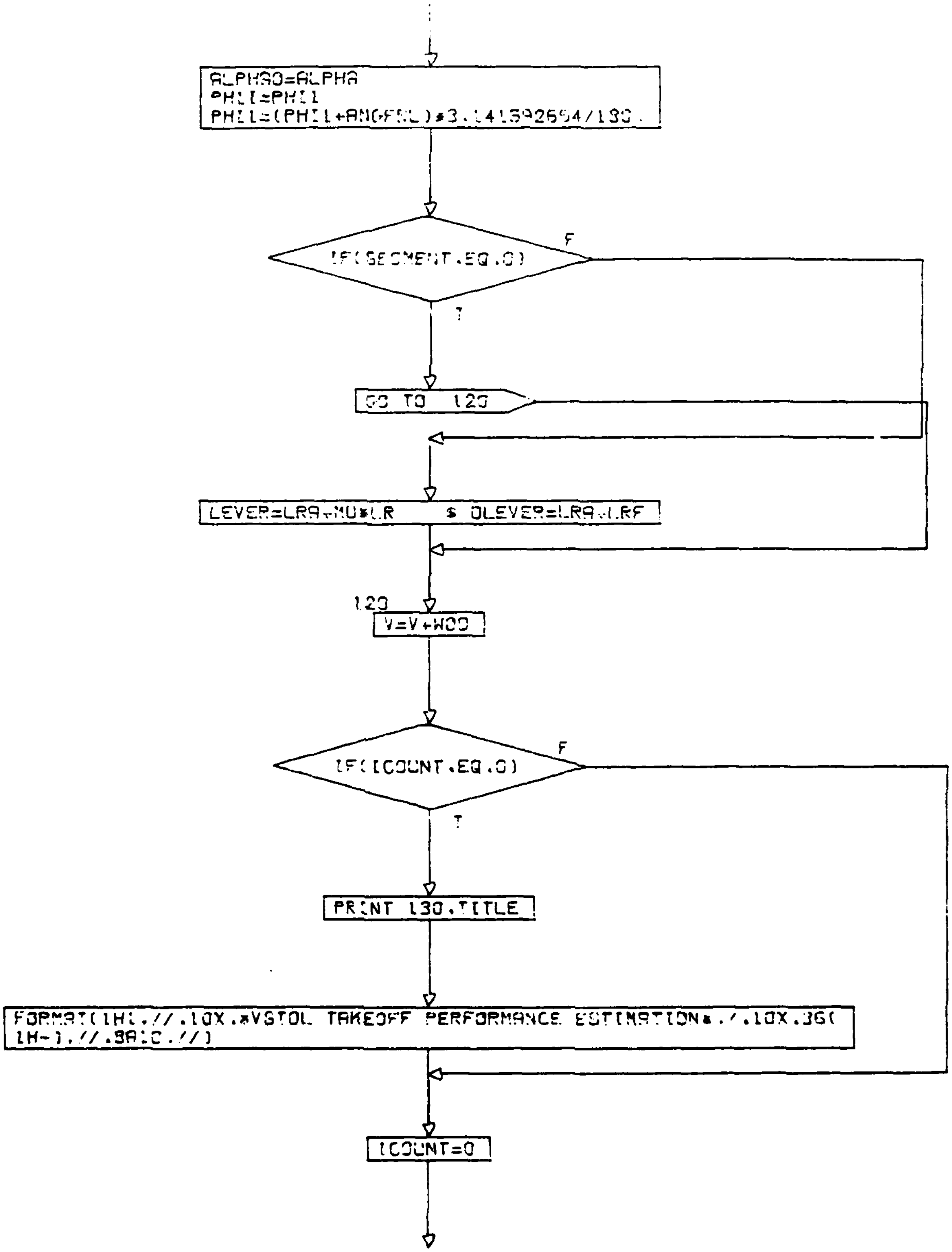
CONT. ON PG 3


IF(EOF(9)) 60,50
≠
0
50
IC=IC+1
WRITE(8,30) ICOPY
PRINT 40,ICOPY
IF(IC.NE.2.AND.ICOPY(2).EQ.1H$)
F
T
GO TO 60
GO TO 20
2
92
60
DO 70 II=1,IC
3
70
BACKSPACE 9
READ DATA
READSC.TITLE

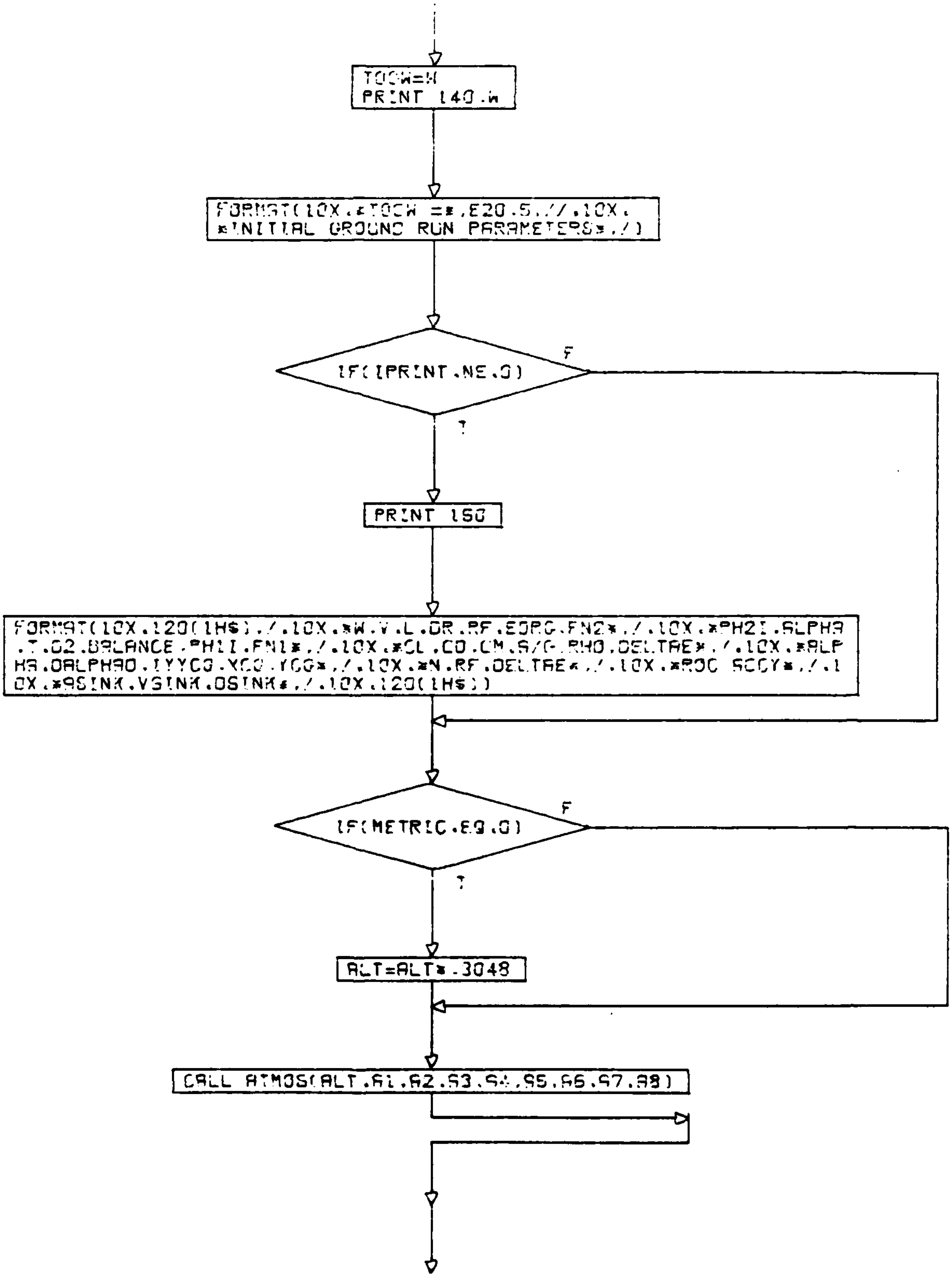
CONT. ON PG 4


CALL TRESS(1,WD,XD,YD,ZD,FWXYZ)
80
FORMAT(8A10)
GO TO 110
2
A3
90
REWIND 8
READ(8,DATA)
READ 30,ICOPY
IF(EOF(8)) 750,100
≠
35
D9
0
100
PRINT 40,ICOPY
BACKSPACE 8 & READ DATA
110
PH21=PHI2

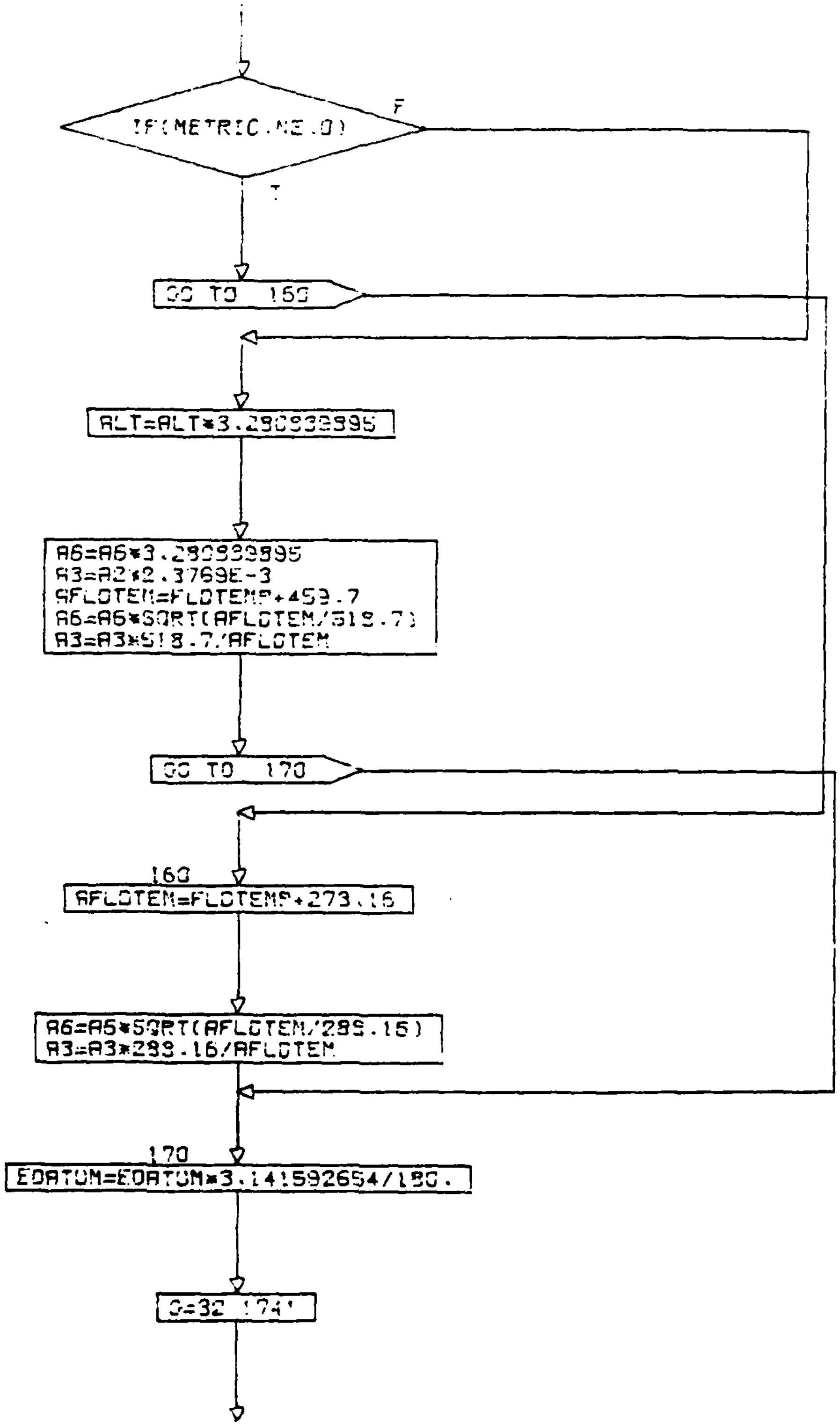
CONT. ON PG 5


ALPHAO=ALPHA
PHII=PHI1
PHI1=(PHI1+ANGFSL)*3.141592654/180.
IF(SEGMENT.EQ.0)
F
T
GO TO 120
LEVER=LRA+MU*LR $ OLEVER=LRA+LRF
120
V=V+WOD
IF(ICOUNT.EQ.0)
F
T
PRINT 130,TITLE
FORMAT(1H1,//,10X,*VSTOL TAKEOFF PERFORMANCE ESTIMATION*,/,10X,36(
1H-),//,8A10,//)
ICOUNT=0

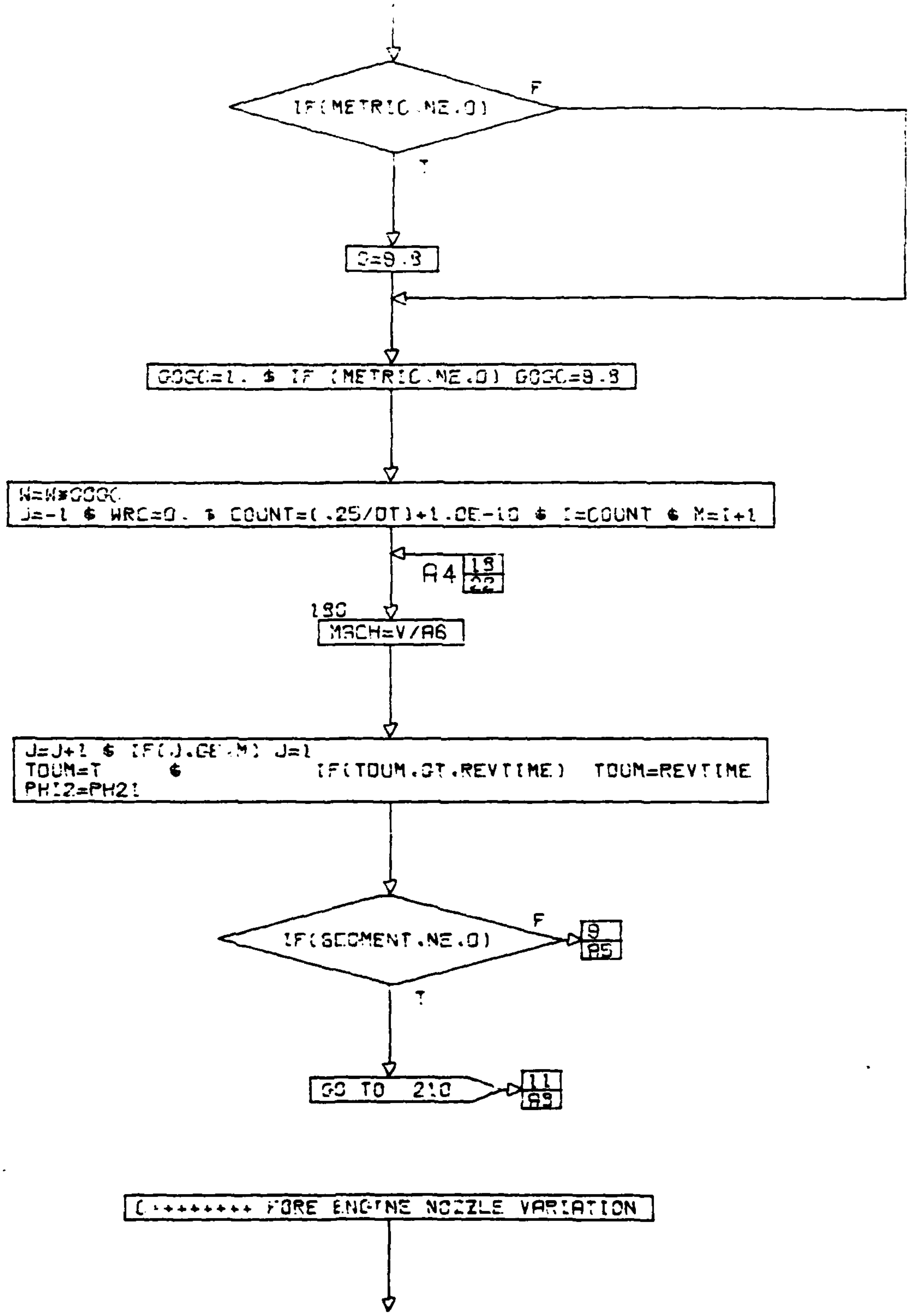
CONT. ON PG 6


TOCW=W
PRINT 140.W
FORMAT(10X.*TOCW =*,E20.5.//.10X.
INITIAL GROUND RUN PARAMETERS./)
IF(IPRINT.NE.0)
F
T
PRINT 150
FORMAT(10X.120(1H$)./.10X.*W.V.L.DR.RF.EDRG.FN2*./.10X.*PH2I.ALPHA
.T.D2.BALANCE.PH1I.FN1*./.10X.*CL.CD.CM.S/G.RHO.DELTAE*./.10X.*ALP
HA.DALPHAD.IYYCG.XCG.YCG*,/.10X.*N.RF.DELTAE*./.10X.*ROC SCCY*./.1
0X.*ASINK.VSINK.DSINK*./.10X.120(1H$))
IF(METRIC.EQ.0)
F
T
ALT=ALT*.3048
CALL ATMOS(ALT.A1.A2.A3.A4.A5.A6.A7.A8)

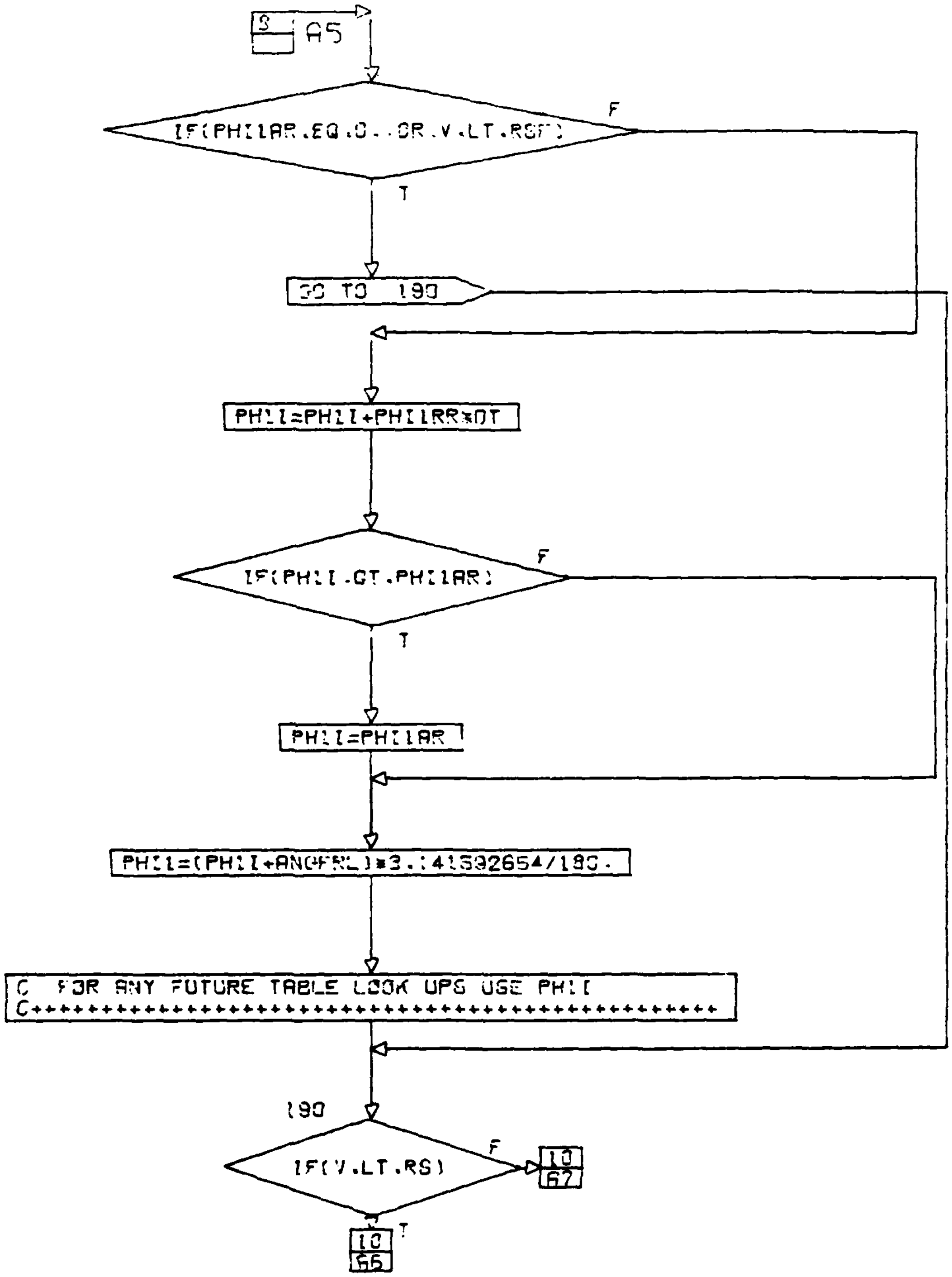
CONT. ON PG 7


IF(METRIC.NE.0)
F
T
GO TO 160
ALT=ALT*3.280839895
A6=A6*3.280839895
A3=A2*2.3769E-3
AFLOTEM=FLOTEMP+459.7
A6=A6*SQRT(AFLOTEM/518.7)
A3=A3*518.7/AFLOTEM
GO TO 170
160
AFLOTEM=FLOTEMP+273.16
A6=A5*SQRT(AFLOTEM/288.15)
A3=A3*288.16/AFLOTEM
170
EDATUM=EDATUM*3.141592654/180.
G=32.174

CONT. ON PG 8

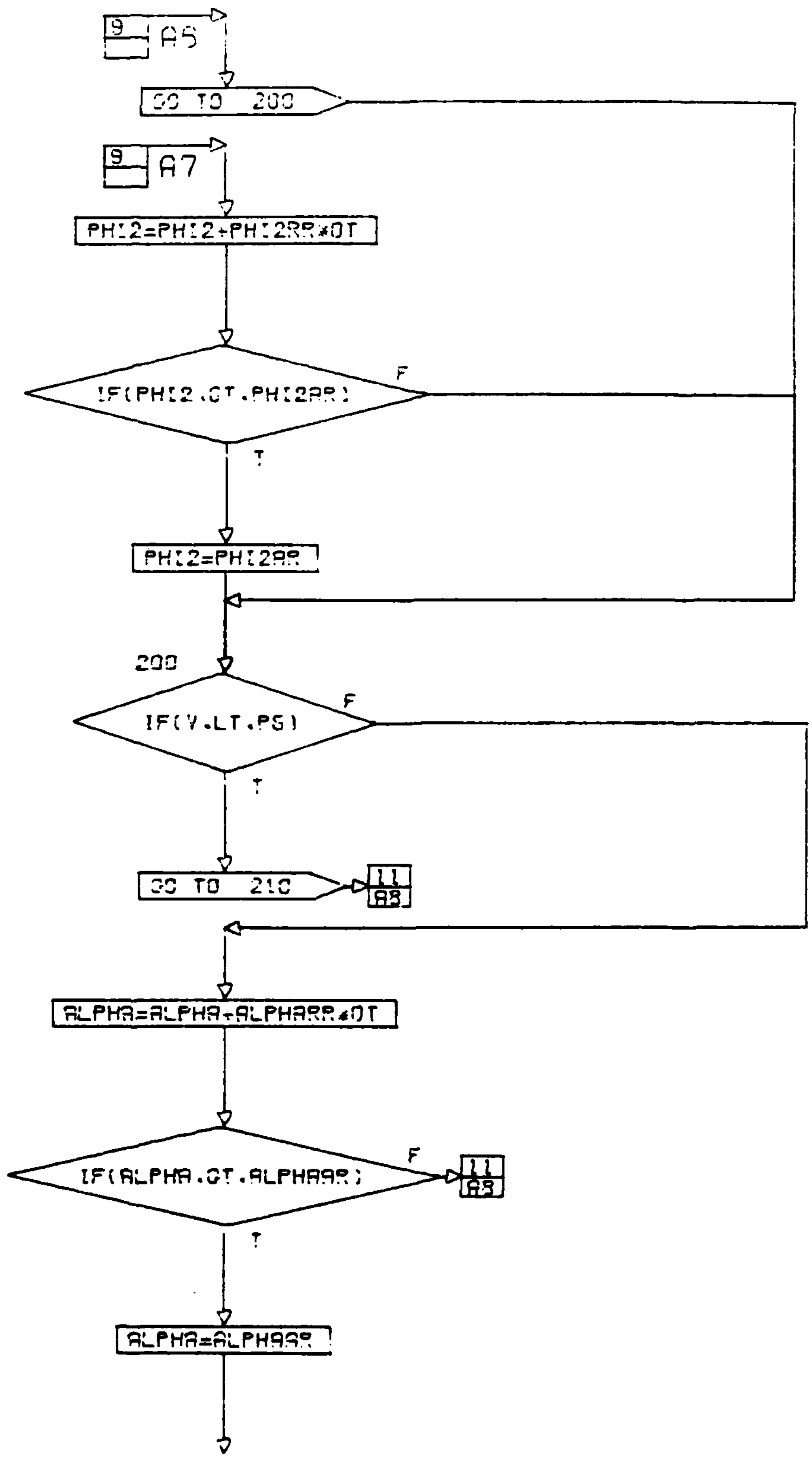

IF(METRIC.NE.0)
F
T
G=9.8
GOGC=1. $ IF (METRIC.NE.0) GOGC=9.8
W=W*GOGC
J=-1 $ WRC=0. $ COUNT=(.25/DT)+1.0E-10 $ I=COUNT $ M=I+1
A4 18/22
180
MACH=V/A6
J=J+1 $ IF(J.GE.M) J=1
TOUM=T $ IF(TOUM.GT.REVTIME) TOUM=REVTIME
PHI2=PH21
IF(SEGMENT.NE.0)
F
9/A5
T
GO TO 210
11/A9
C++++++++ FORE ENGINE NOZZLE VARIATION

CONT. ON PG 9

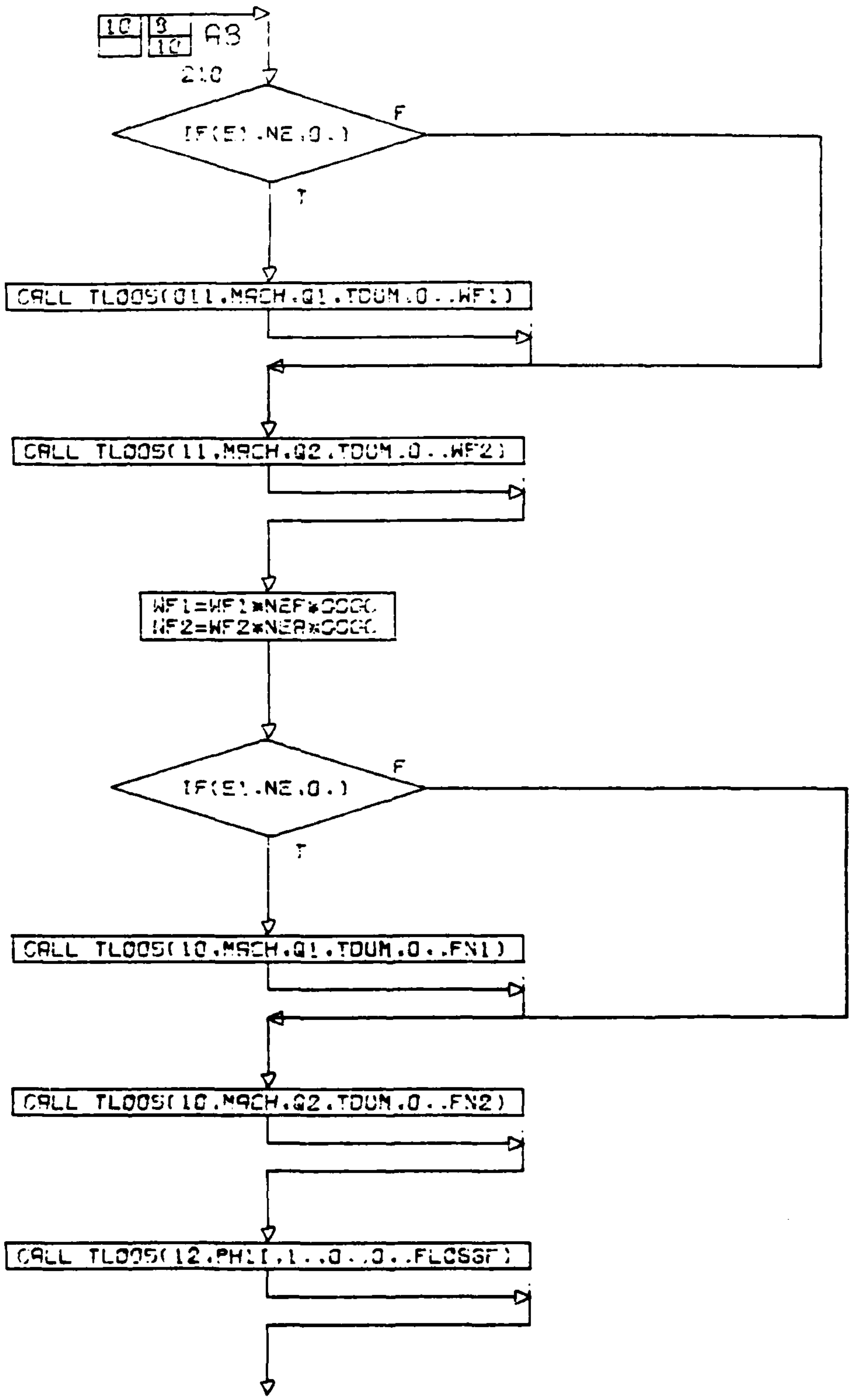


3
A5
IF(PHI1AR.EQ.0..OR.V.LT.RS?)
F
T
GO TO 190
PHI1=PHI1+PHI1RR*DT
IF(PHI1.GT.PHI1AR)
F
T
PHI1=PHI1AR
PHI1=(PHI1+ANGFRL)*3.141592654/180.
C FOR ANY FUTURE TABLE LOOK UPS USE PHI1
C++
190
IF(V.LT.RS)
F
10
67
T
10
66

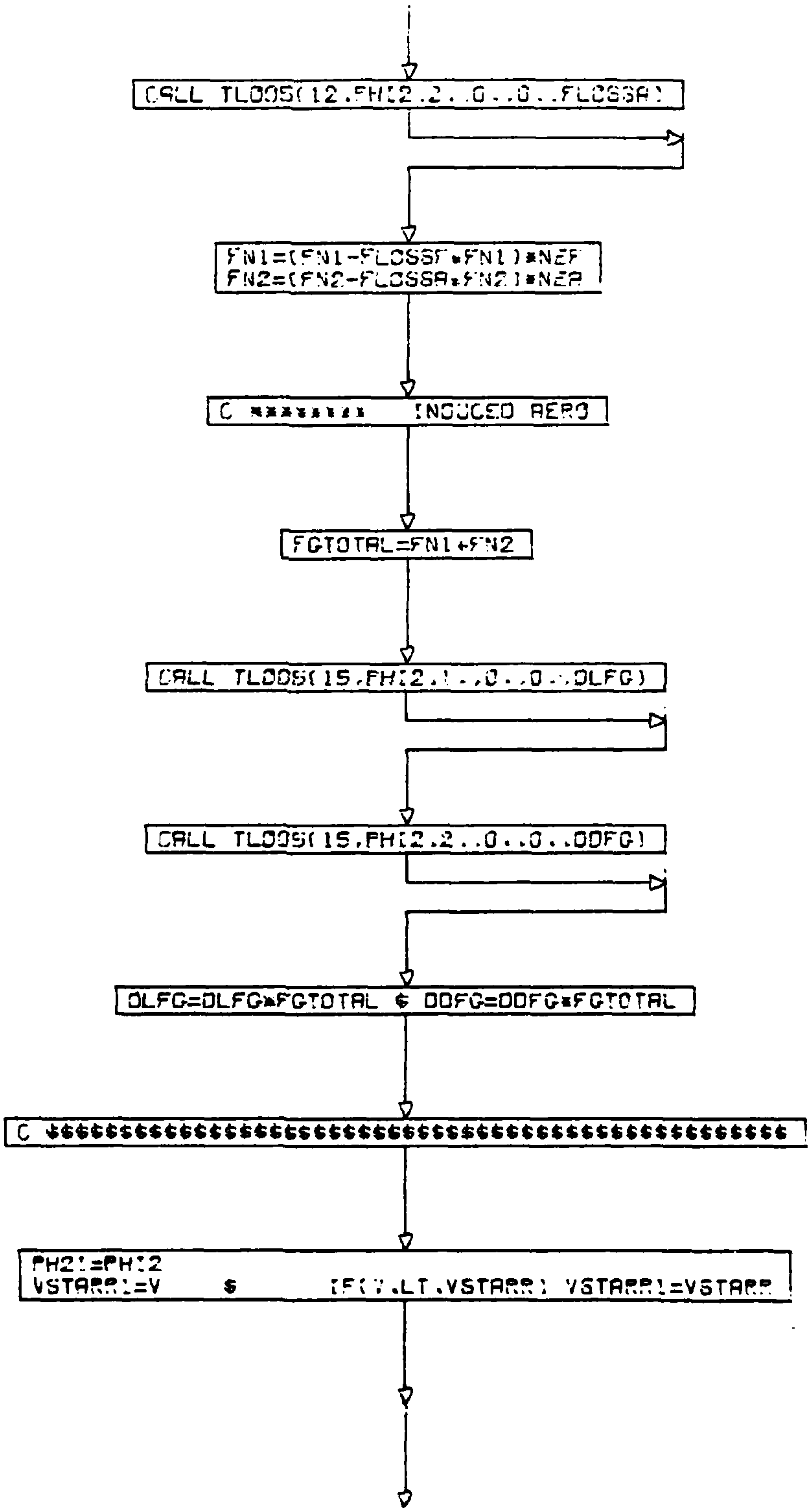
CONT. ON PG 10


9
A5
GO TO 200
9
A7
PHI2=PHI2+PHI2RR*DT
IF(PHI2.GT.PHI2AR)
F
T
PHI2=PHI2AR
200
IF(V.LT.PS)
F
T
GO TO 210
11
A8
ALPHA=ALPHA+ALPHARR*DT
IF(ALPHA.GT.ALPHAAR)
F
11
A8
T
ALPHA=ALPHAAR

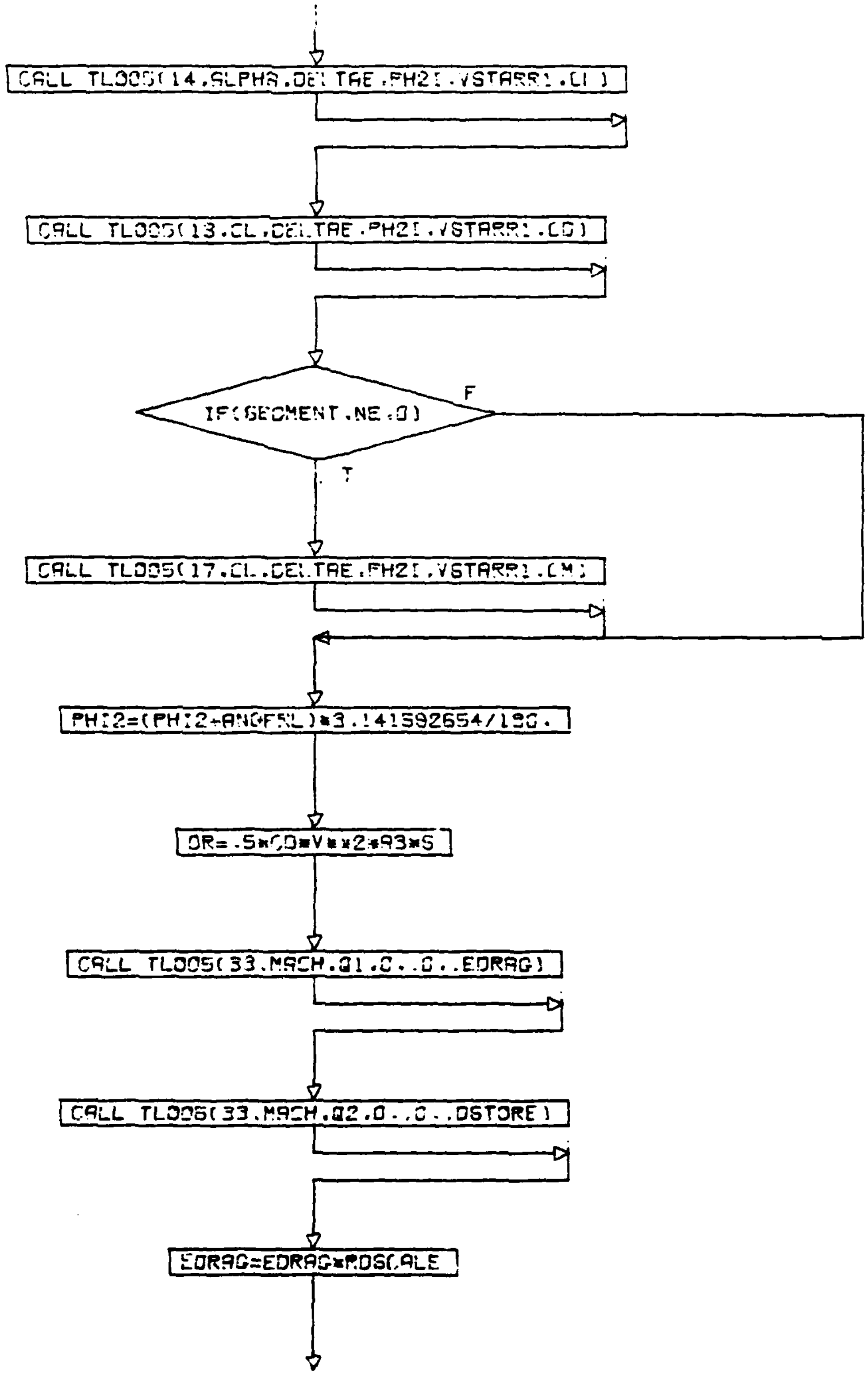
CONT. ON PG 11


10
8
10
A3
210
IF(E1.NE.0.)
F
T
CALL TLOOS(011.MACH.Q1.TOUM.0..WF1)
CALL TLOOS(11.MACH.Q2.TOUM.0..WF2)
WF1=WF1*NEF*CCCC
WF2=WF2*NEA*CCCC
IF(E1.NE.0.)
F
T
CALL TLOOS(10.MACH.Q1.TOUM.0..FN1)
CALL TLOOS(10.MACH.Q2.TOUM.0..FN2)
CALL TLOOS(12.PHII,1..0..0..FLOSSF)

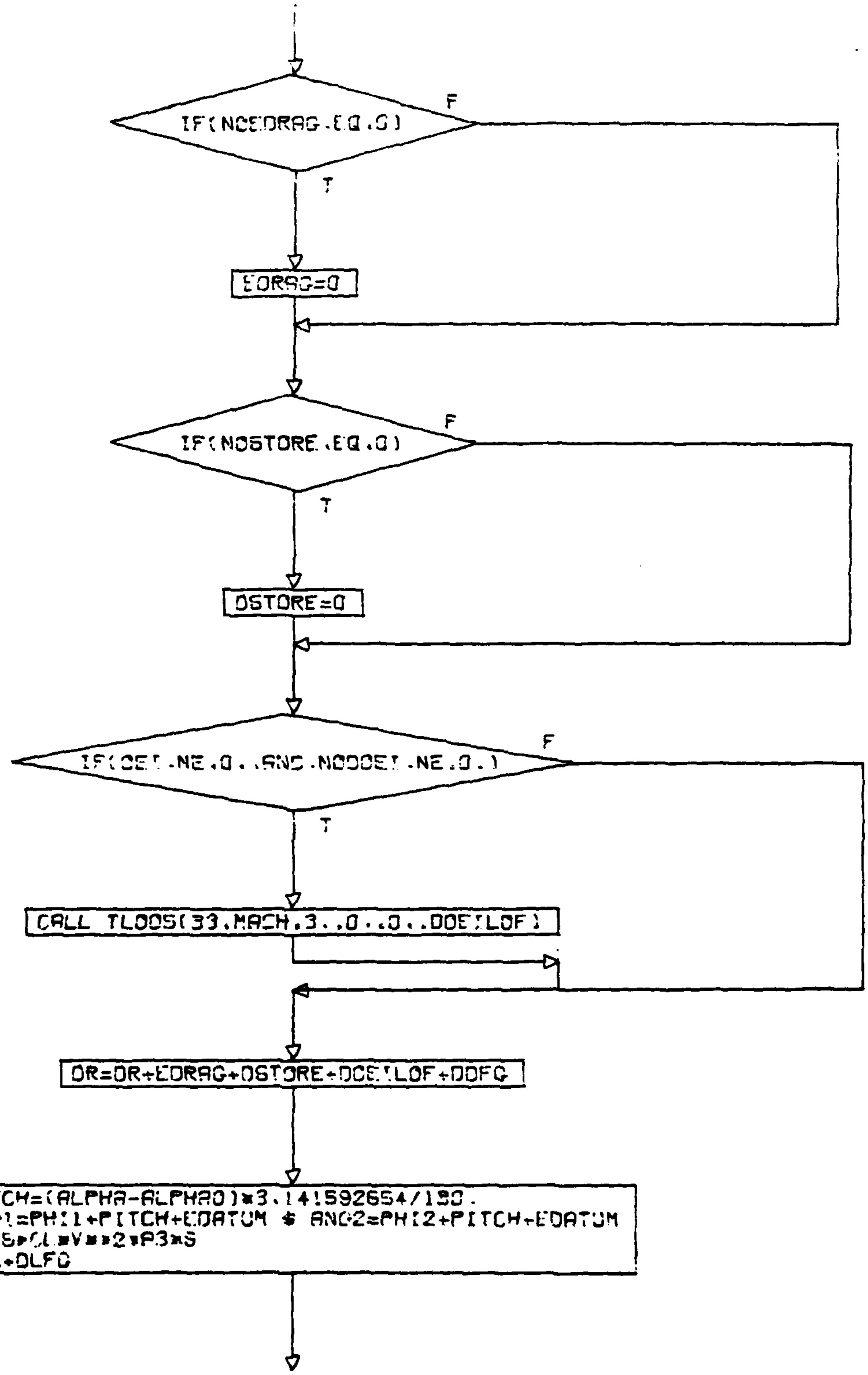
CONT. ON PG 12


CALL TLOOS(12.PHI2.2..0..0..FLOSSA)
FN1=(FN1-FLOSSF*FN1)*NEF
FN2=(FN2-FLOSSA*FN2)*NEA
C ********* INDUCED AERO
FGTOTAL=FN1+FN2
CALL TLOOS(15,PHI2.1..0..0..DLFG)
CALL TLOOS(15,PHI2.2..0..0..DDFG)
DLFG=DLFG*FGTOTAL $ DDFG=DDFG*FGTOTAL
C $$
PH21=PHI2
VSTARR1=V $ IF(V.LT.VSTARR) VSTARR1=VSTARR

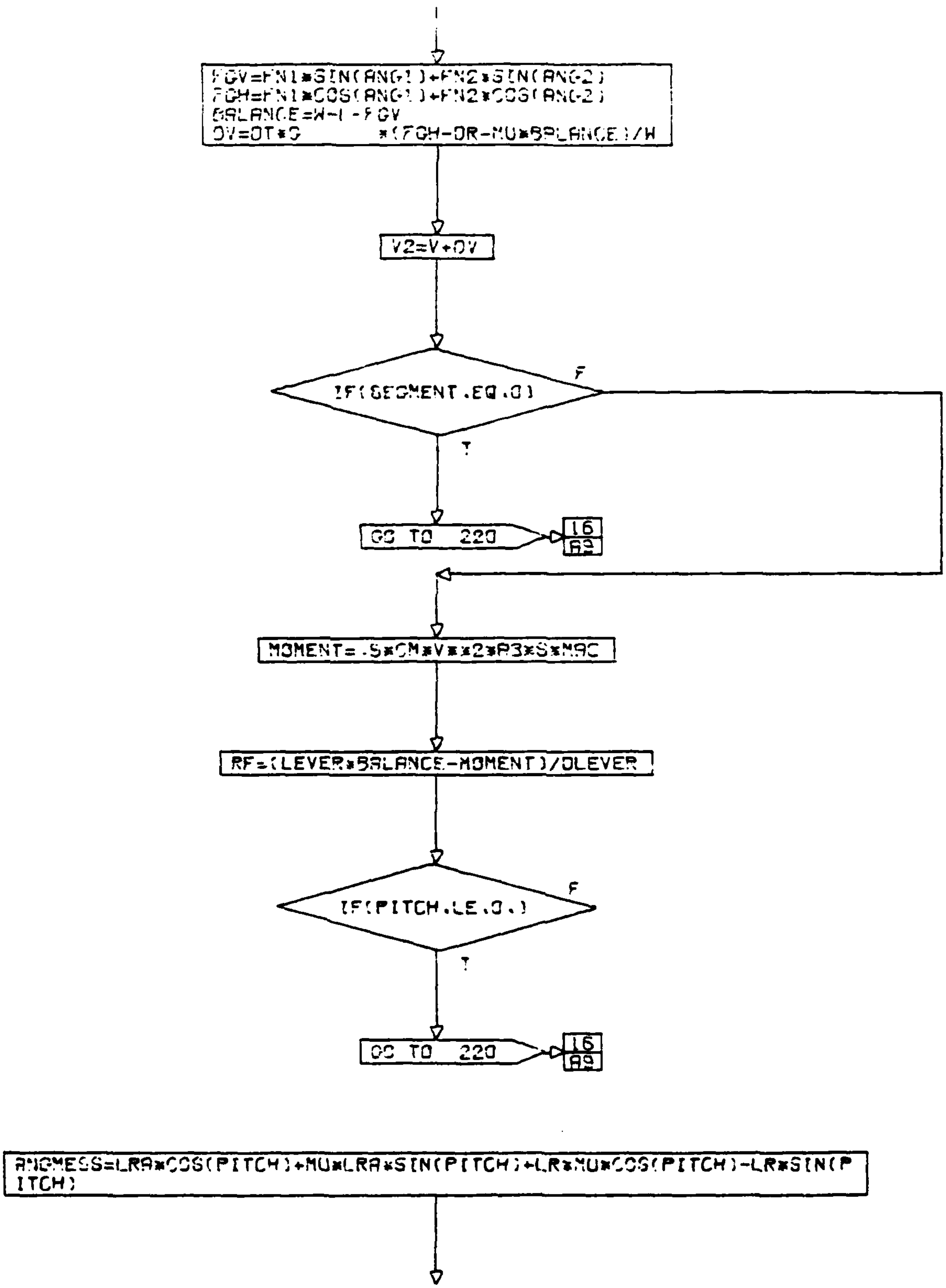
CONT. ON PG 13


CALL TLOO5(14.ALPHA.DELTAE.PHZI.VSTARR1.CL)
CALL TLOO5(13.CL.DELTAE.PHZI.VSTARR1.CD)
IF(GEOMENT.NE.0)
F
T
CALL TLOO5(17.CL.DELTAE.PHZI.VSTARR1.CM)
PHI2=(PHI2+ANGFRL)*3.141592654/180.
DR=.5*CD*V**2*A3*S
CALL TLOO5(33.MACH.Q1.0..0..EDRAG)
CALL TLOO6(33.MACH.Q2.0..0..DSTORE)
EDRAG=EDRAG*RDSCALE

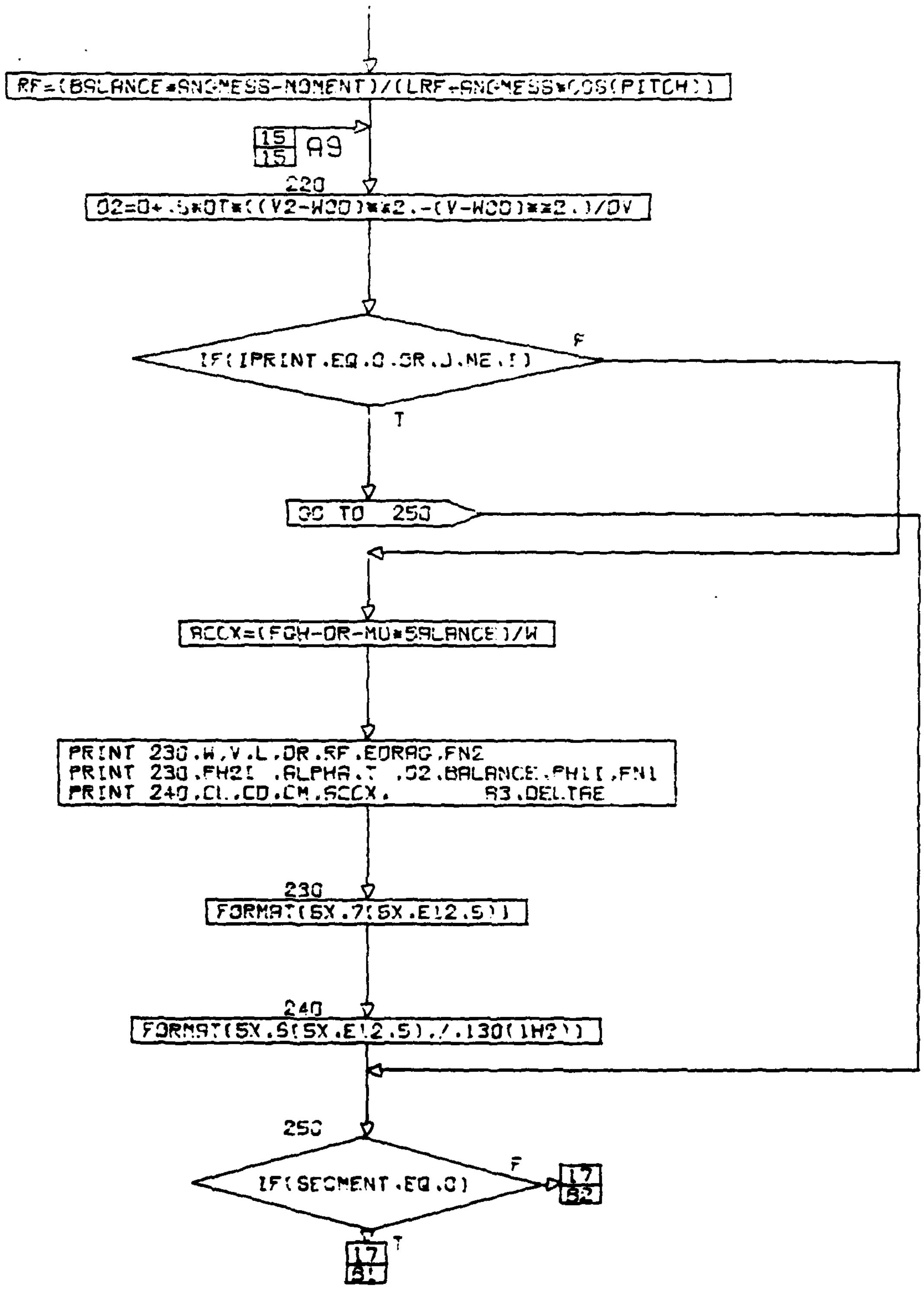
CONT. ON PG 14


IF(NCEDRAG.EQ.0)
F
T
EDRAG=0
IF(NOSTORE.EQ.0)
F
T
DSTORE=0
IF(DCET.NE.0..AND.NODCET.NE.0.)
F
T
CALL TLOOS(33,MACH,3.,0.,0.,DCETLOF)
DR=DR+EDRAG+DSTORE+DCETLOF+DDFG
PITCH=(ALPHA-ALPHA0)*3.141592654/180.
ANG1=PHI1+PITCH+EDATUM $ ANG2=PHI2+PITCH+EDATUM
L=.5*CL*V**2*P3*S
L=L+DLFG

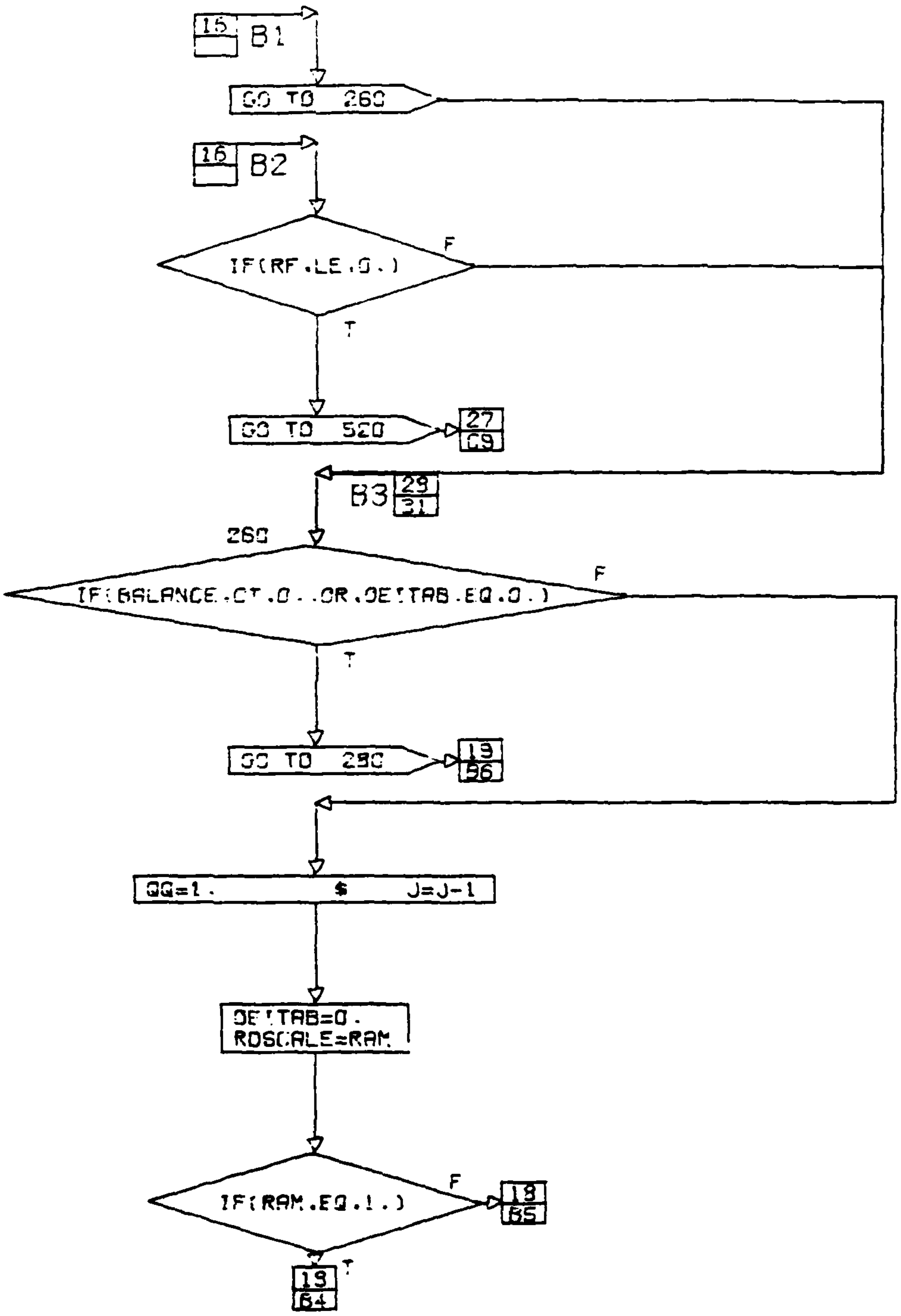
CONT. ON PG 15


FGV=FN1*SIN(ANG1)+FN2*SIN(ANG2)
FGH=FN1*COS(ANG1)+FN2*COS(ANG2)
BALANCE=W-L-FGV
DV=DT*G *(FGH-DR-MU*BALANCE)/W
V2=V+DV
IF(SEGMENT.EQ.0)
F
T
GO TO 220
16
A9
MOMENT=.5*CM*V**2*R3*S*MAC
RF=(LEVER*BALANCE-MOMENT)/DLEVER
IF(PITCH.LE.0.)
F
T
GO TO 220
16
A9
ANGMESS=LRA*COS(PITCH)+MU*LRA*SIN(PITCH)+LR*MU*COS(PITCH)-LR*SIN(P
ITCH)

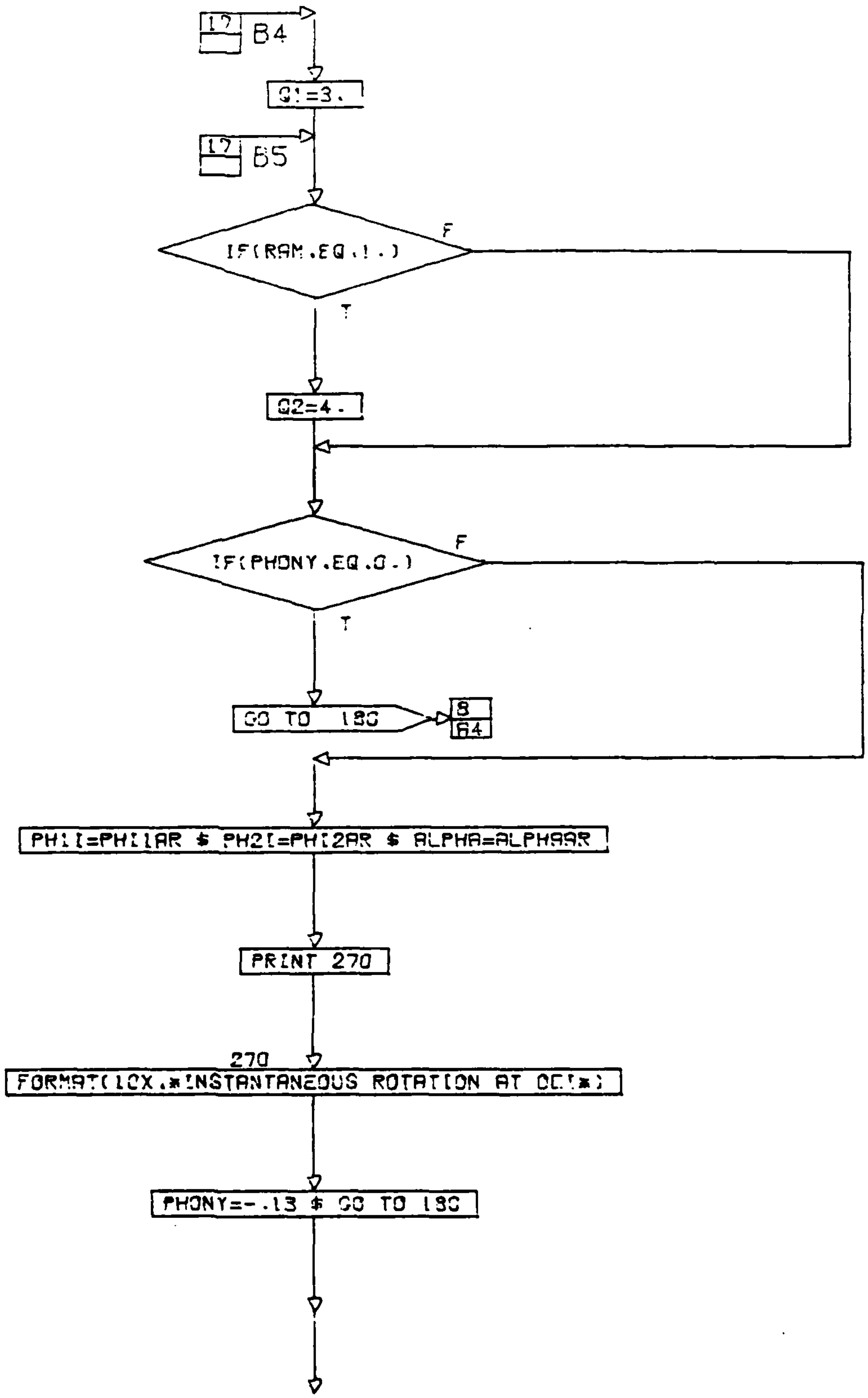
CONT. ON PG 16


RF=(BALANCE*ANGMESS-MOMENT)/(LRF+ANGMESS*COS(PITCH))
15
15
A9
220
D2=D+.5*DT*((V2-WOD)**2.-(V-WOD)**2.)/DV
IF(IPRINT.EQ.0.OR.J.NE.I)
F
T
GO TO 250
ACCX=(FGH-DR-MU*BALANCE)/W
PRINT 230,W,V,L,DR,RF,EDRAG,FN2
PRINT 230,FH2I ,ALPHA,T ,D2,BALANCE,PHII,FN1
PRINT 240,CL,CD,CM,ACCX, A3,DELTAE
230
FORMAT(5X,7(5X,E12.5))
240
FORMAT(5X,5(5X,E12.5),/,130(1H?))
250
IF(SEGMENT.EQ.0)
F
17
B2
T
17
B1

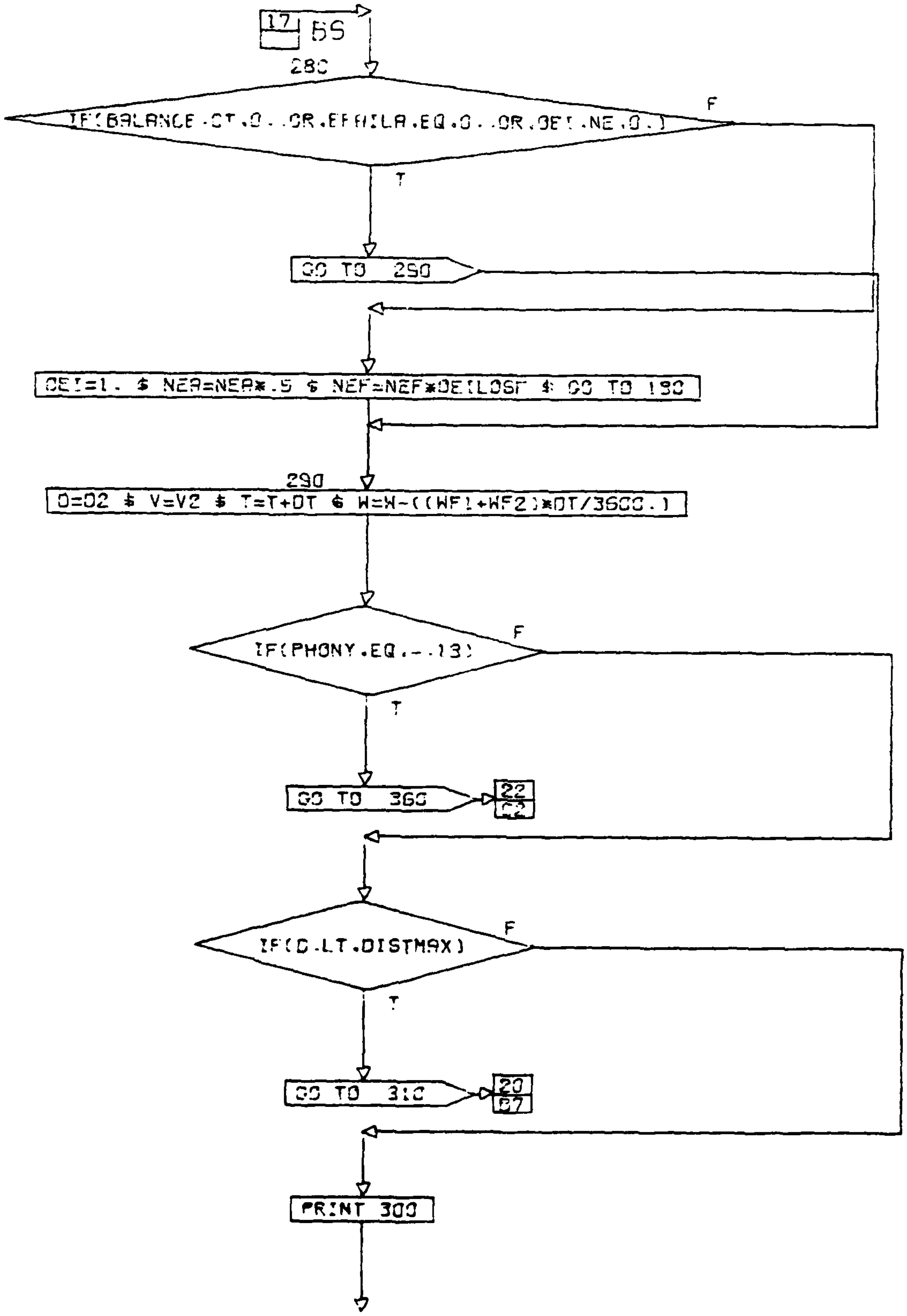
CONT. ON PG 17


15
B1
GO TO 260
16
B2
IF(RF.LE.0.)
F
T
GO TO 520
27
C9
B3
29
31
260
IF(BALANCE.GT.0..OR.DEITAB.EQ.0.)
F
T
GO TO 280
19
B6
GG=1. $ J=J-1
DEITAB=0.
ROSCALE=RAM
IF(RAM.EQ.1.)
F
18
B5
T
19
B4

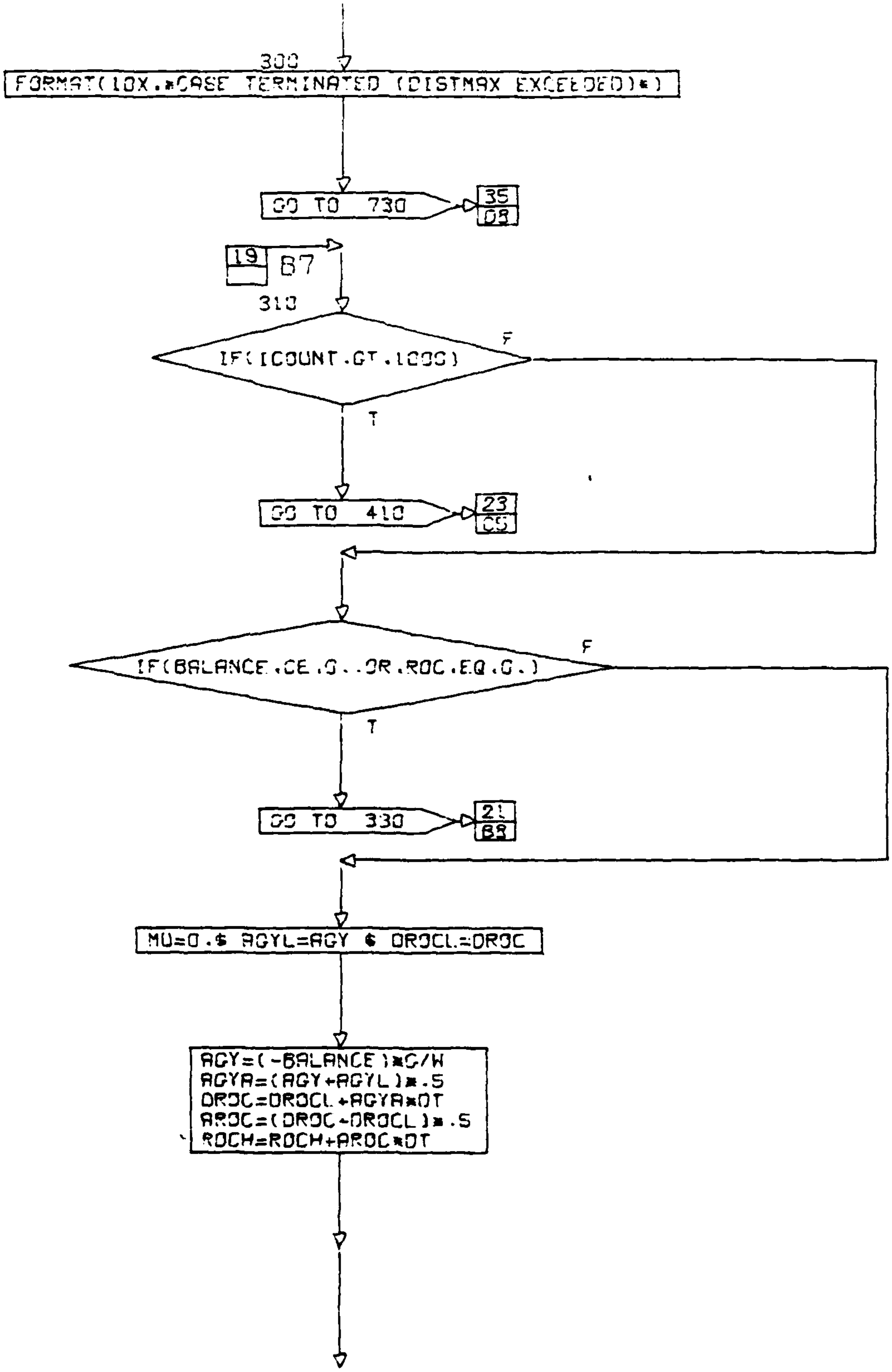
CONT. ON PG 19


17
B4
Q1=3.
17
B5
IF(RAM.EQ.1.)
F
T
Q2=4.
IF(PHONY.EQ.0.)
F
T
GO TO 180
8
B4
PH1I=PH1IAR $ PH2I=PH12AR $ ALPHA=ALPHAAR
PRINT 270
270
FORMAT(10X,*INSTANTANEOUS ROTATION AT OC!*)
PHONY=-.13 $ GO TO 180

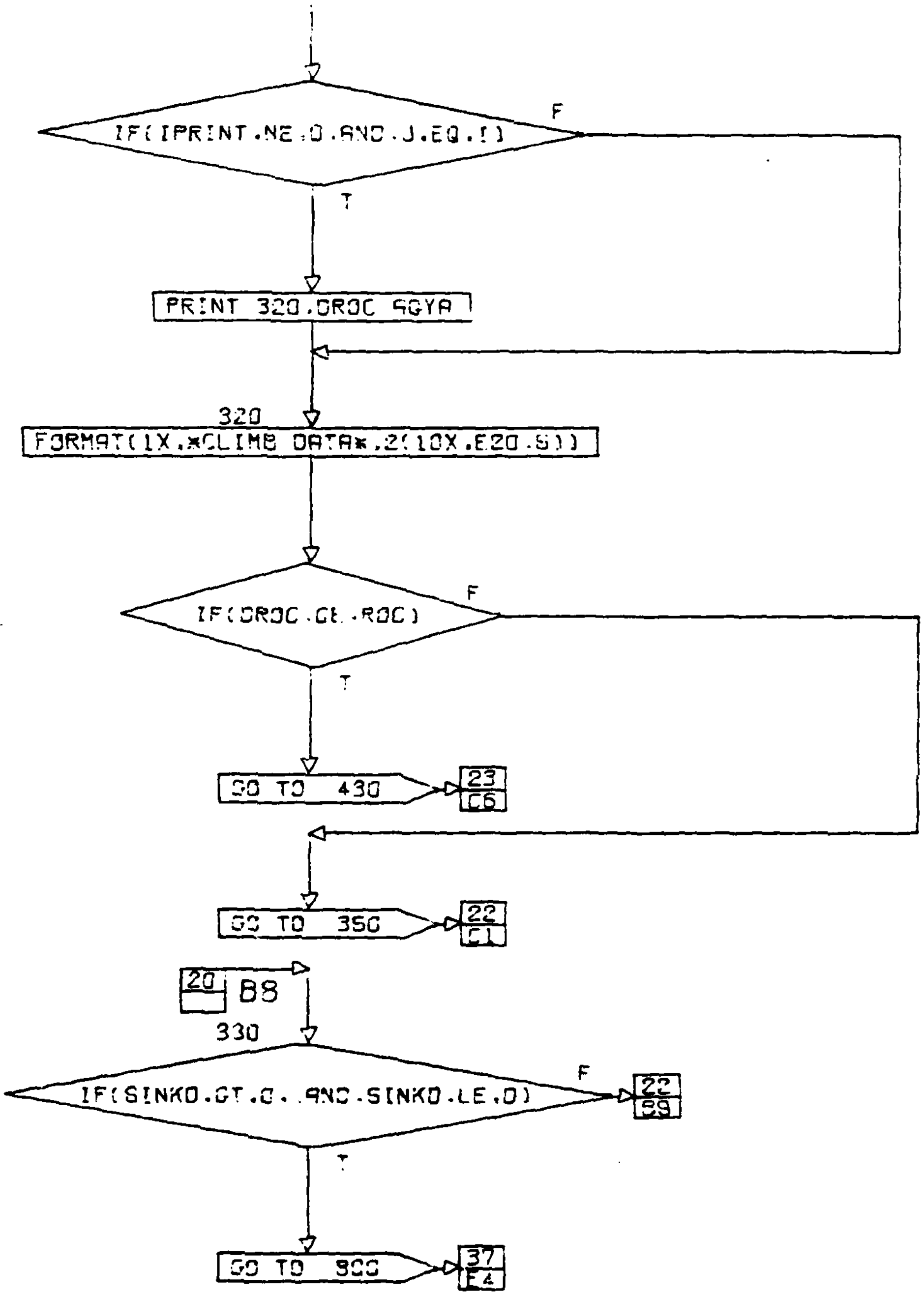
CONT. ON PG 19


17
65
280
IF(BALANCE.GT.0..OR.EFAILA.EQ.0..OR.OEI.NE.0.)
F
T
GO TO 290
OEI=1. $ NEA=NEA*.5 $ NEF=NEF*OEILOSF $ GO TO 190
290
D=D2 $ V=V2 $ T=T+DT $ W=W-((WF1+WF2)*DT/3600.)
IF(PHONY.EQ.-.13)
F
T
GO TO 360
22
62
IF(D.LT.DISTMAX)
F
T
GO TO 310
20
67
PRINT 300

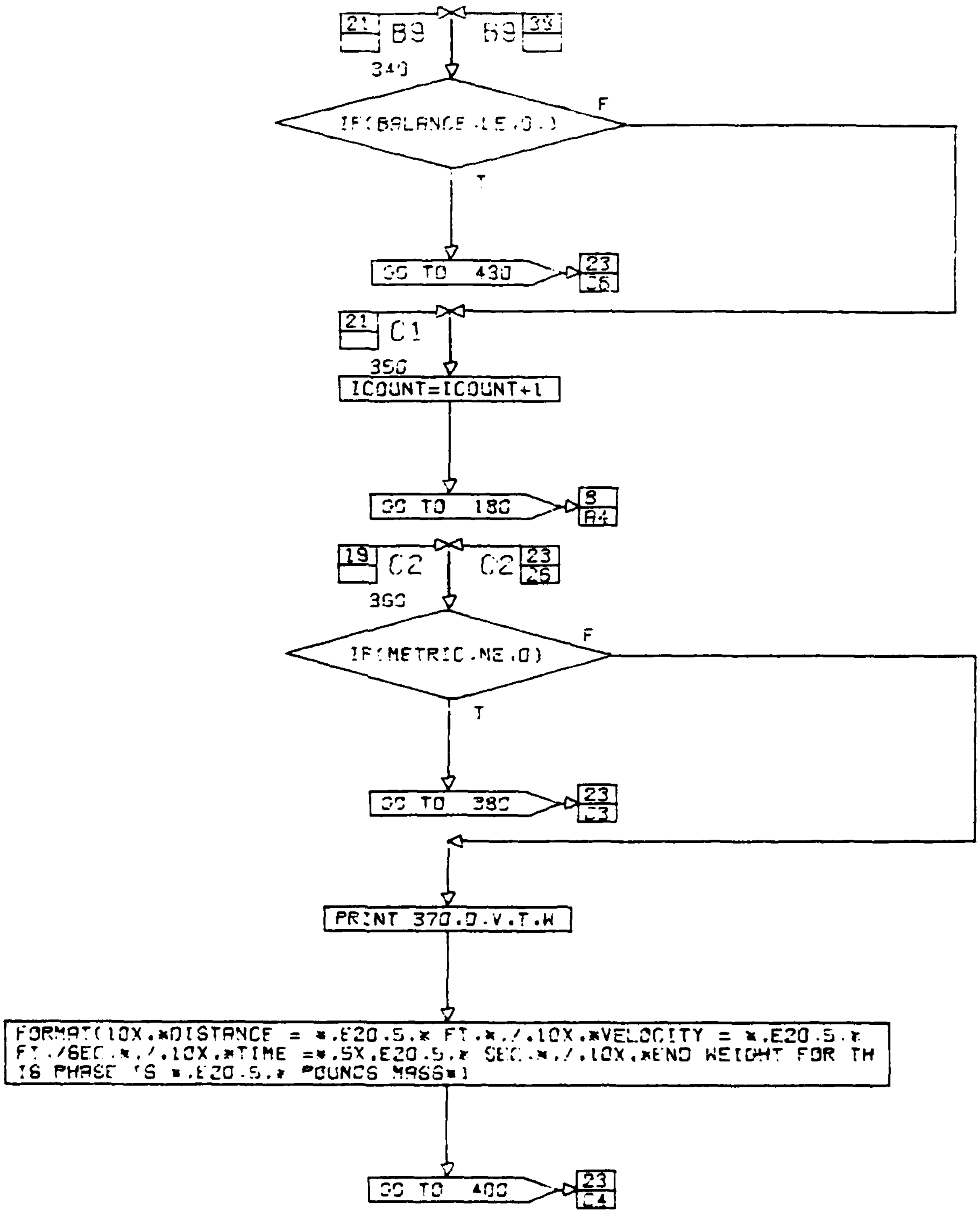
CONT. ON PG 20


300
FORMAT(10X,*CASE TERMINATED (DISTMAX EXCEEDED)*)
GO TO 730
35
D8
19
B7
310
IF(ICOUNT.GT.1000)
F
T
GO TO 410
23
C5
IF(BALANCE.GE.0..OR.ROC.EQ.0.)
F
T
GO TO 330
21
B8
MU=0.$ AGYL=AGY $ DROCL=DROC
AGY=(-BALANCE)*G/W
AGYA=(AGY+AGYL)*.5
DROC=DROCL+AGYA*DT
AROC=(DROC+DROCL)*.5
ROCH=ROCH+AROC*DT

CONT. ON PG 21

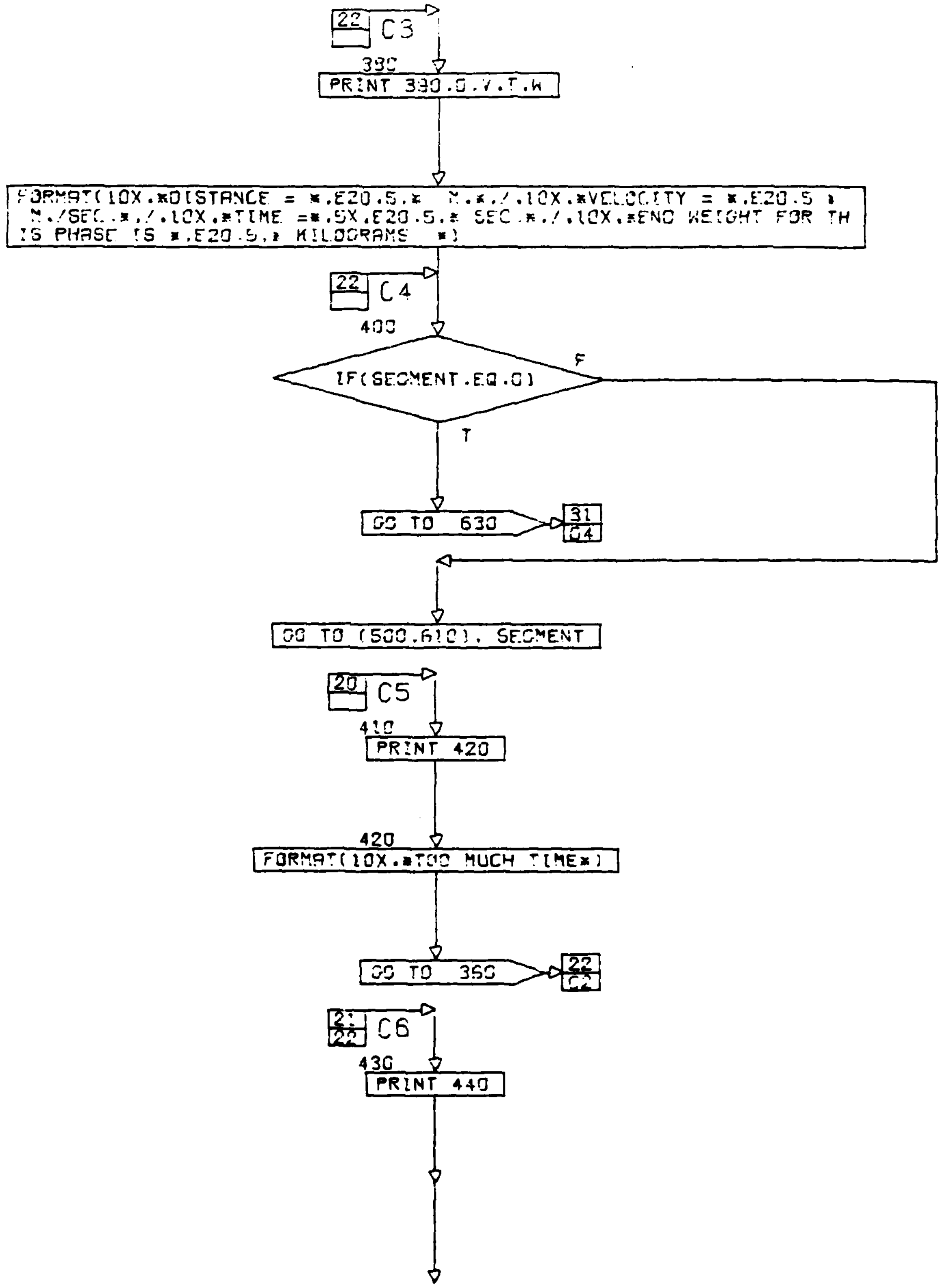

IF(IPRINT.NE.0.AND.J.EQ.1)
F
T
PRINT 320,DROC AGYA
320
FORMAT(1X,*CLIMB DATA*,2(10X,E20.8))
IF(DROC.GE.ROC)
F
T
GO TO 430
23
C6
GO TO 350
22
C1
20
B8
330
IF(SINKO.GT.0..AND.SINKO.LE.D)
F
22
59
T
GO TO 800
37
F4

CONT. ON PG 22

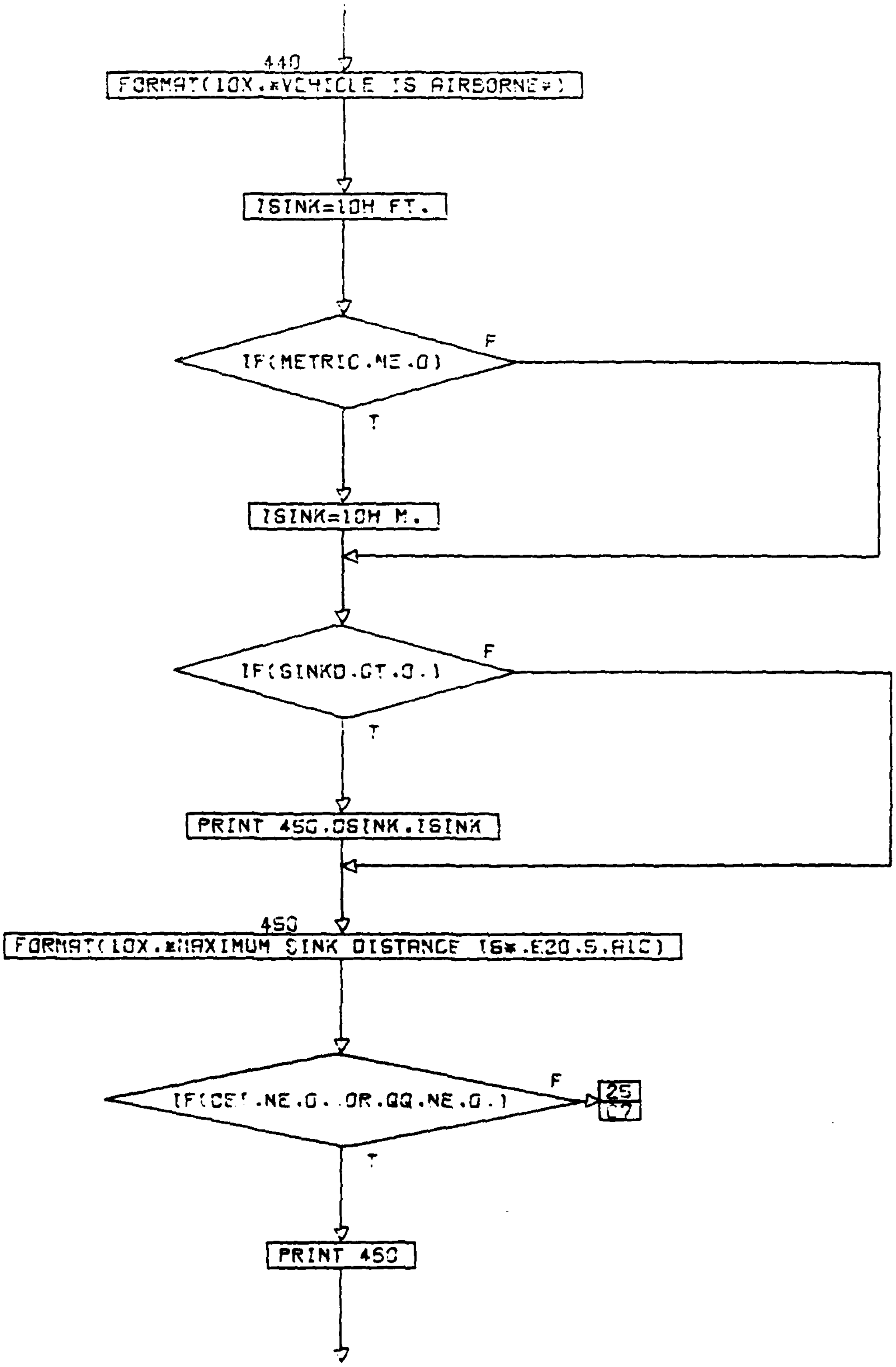


21 B9
B9 39
340
IF(BALANCE.LE.0.)
F
T
GO TO 430
23
C6
21 C1
350
ICOUNT=ICOUNT+1
GO TO 180
8
A4
19 C2
C2 23
26
360
IF(METRIC.NE.0)
F
T
GO TO 380
23
C3
PRINT 370,D,V,T,W
FORMAT(10X,*DISTANCE = *,E20.5,* FT.*,/,10X,*VELOCITY = *,E20.5,* FT./SEC.*,/,10X,*TIME =*,5X,E20.5,* SEC.*,/,10X,*END WEIGHT FOR THIS PHASE IS *,E20.5,* POUNDS MASS*)
GO TO 400
23
C4

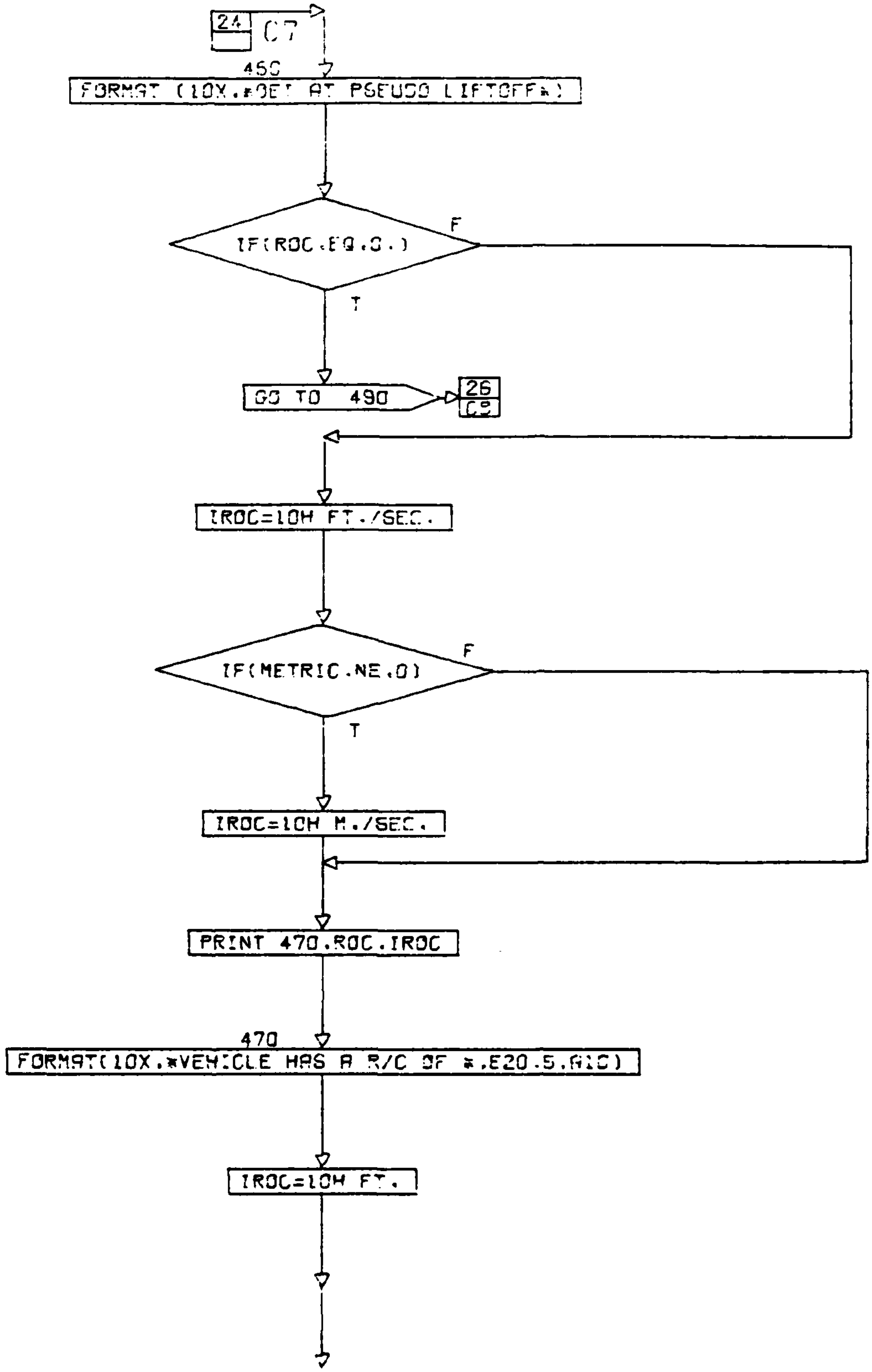
CONT. ON PG 23


22
C3
330
PRINT 330,S,V,T,W
FORMAT(10X,*DISTANCE = *,E20.5,* M.*,/,10X,*VELOCITY = *,E20.5,* M./SEC.*,/,10X,*TIME =*,5X,E20.5,* SEC.*,/,10X,*END WEIGHT FOR THIS PHASE IS *,E20.5,* KILOGRAMS *)
22
C4
400
IF(SEGMENT.EQ.0)
F
T
GO TO 630
31
C4
GO TO (500,610), SEGMENT
20
C5
410
PRINT 420
420
FORMAT(10X,*TOO MUCH TIME*)
GO TO 350
22
C2
C6
430
PRINT 440

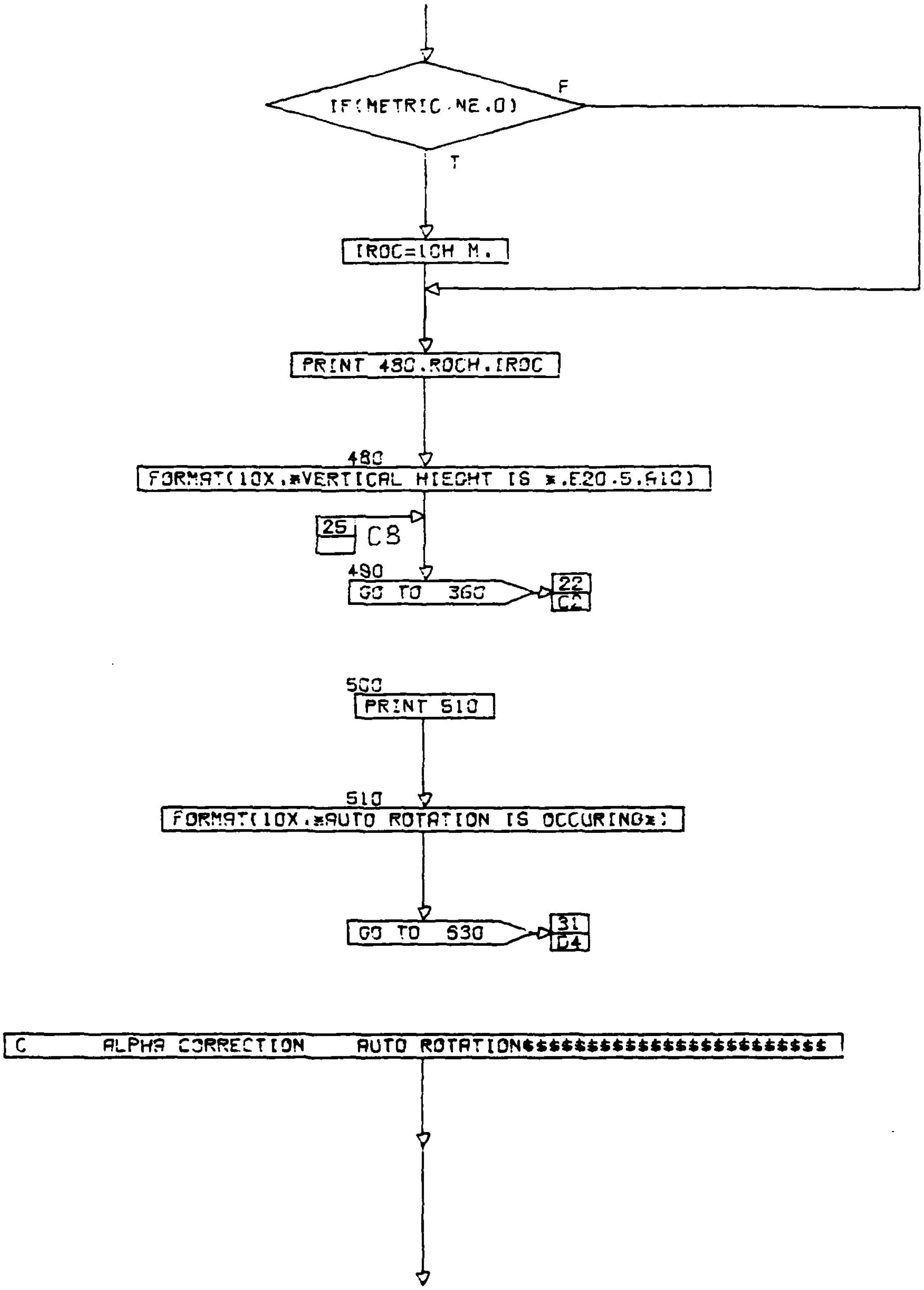
CONT. ON PG 24


440
FORMAT(10X,*VEHICLE IS AIRBORNE*)
ISINK=10H FT.
IF(METRIC.NE.0)
F
T
ISINK=10H M.
IF(SINKD.GT.0.)
F
T
PRINT 450,DSINK,ISINK
450
FORMAT(10X,*MAXIMUM SINK DISTANCE (6*,E20.5,A10)
F
25
T
PRINT 450

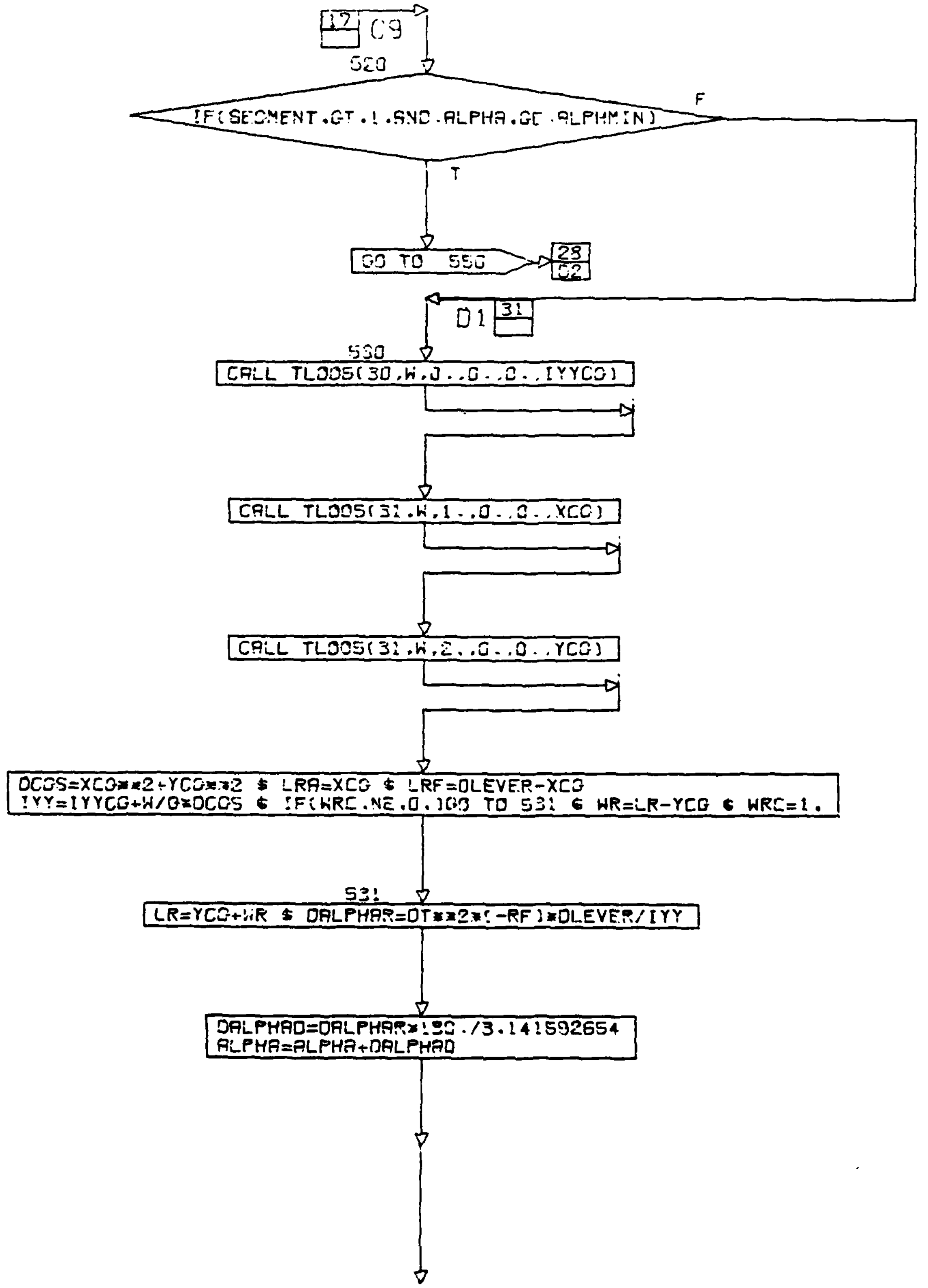
CONT. ON PG 25


24
C7
450
FORMAT (10X,*OET AT PSEUDO LIFTOFF*)
IF(ROC.EQ.0.)
F
T
GO TO 490
26
C8
IROC=10H FT./SEC.
IF(METRIC.NE.0)
F
T
IROC=10H M./SEC.
PRINT 470,ROC,IROC
470
FORMAT(10X,*VEHICLE HAS A R/C OF *,E20.5,A10)
IROC=10H FT.

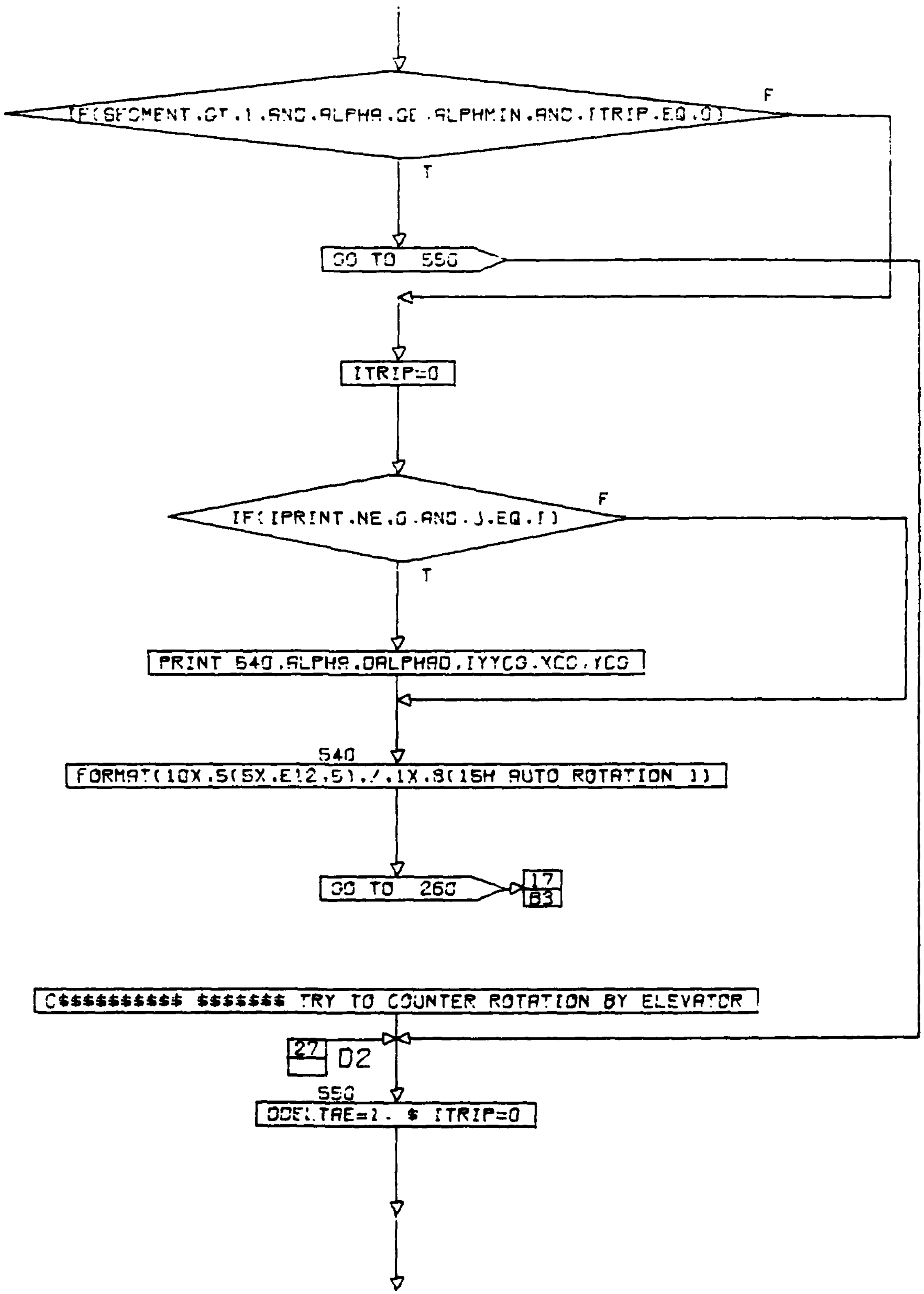
CONT. ON PG 26


IF(METRIC.NE.0)
F
T
IROC=1CH M.
PRINT 480.ROCH.IROC
480
FORMAT(10X,*VERTICAL HIEGHT IS *.E20.5.A10)
25
C8
490
GO TO 360
22
C2
500
PRINT 510
510
FORMAT(10X.*AUTO ROTATION IS OCCURING*)
GO TO 630
31
D4
C ALPHA CORRECTION AUTO ROTATION$$$$$$$$$$$$$$$$$$$$$$$$$

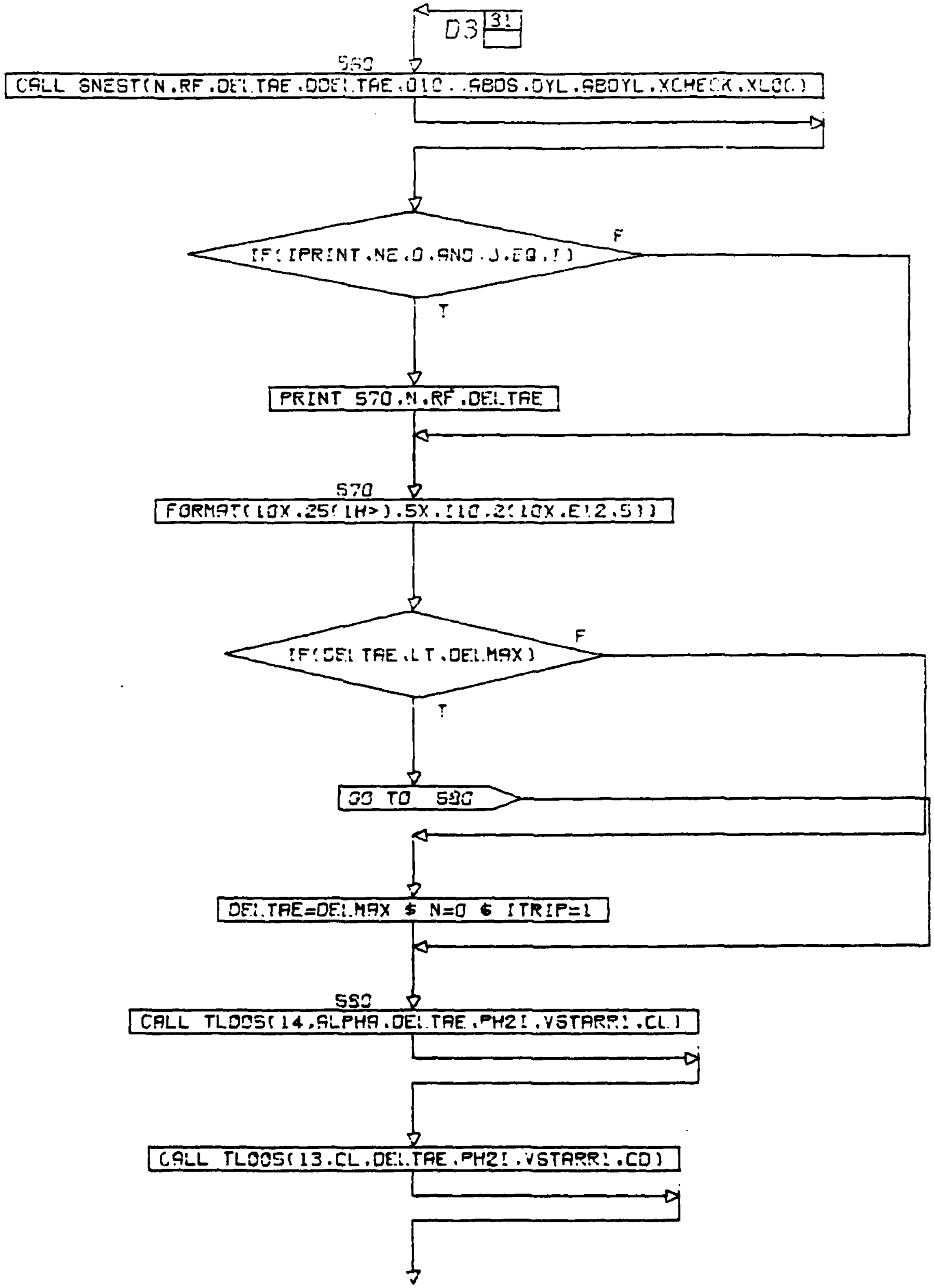
CONT. ON PG 27


17
C9
520
IF(SEGMENT.GT.1.AND.ALPHA.GE.ALPHMIN)
F
T
GO TO 550
28
D2
D1
31
530
CALL TLOOS(30,W,0.,0.,0.,IYYCG)
CALL TLOOS(31,W,1.,0.,0.,XCG)
CALL TLOOS(31,W,2.,0.,0.,YCG)
DCGS=XCG**2+YCG**2 $ LRA=XCG $ LRF=DLEVER-XCG
IYY=IYYCG+W/G*DCGS $ IF(WRC.NE.0.)GO TO 531 $ WR=LR-YCG $ WRC=1.
531
LR=YCG+WR $ DALPHAR=DT**2*(-RF)*DLEVER/IYY
DALPHAD=DALPHAR*180./3.141592654
ALPHA=ALPHA+DALPHAD

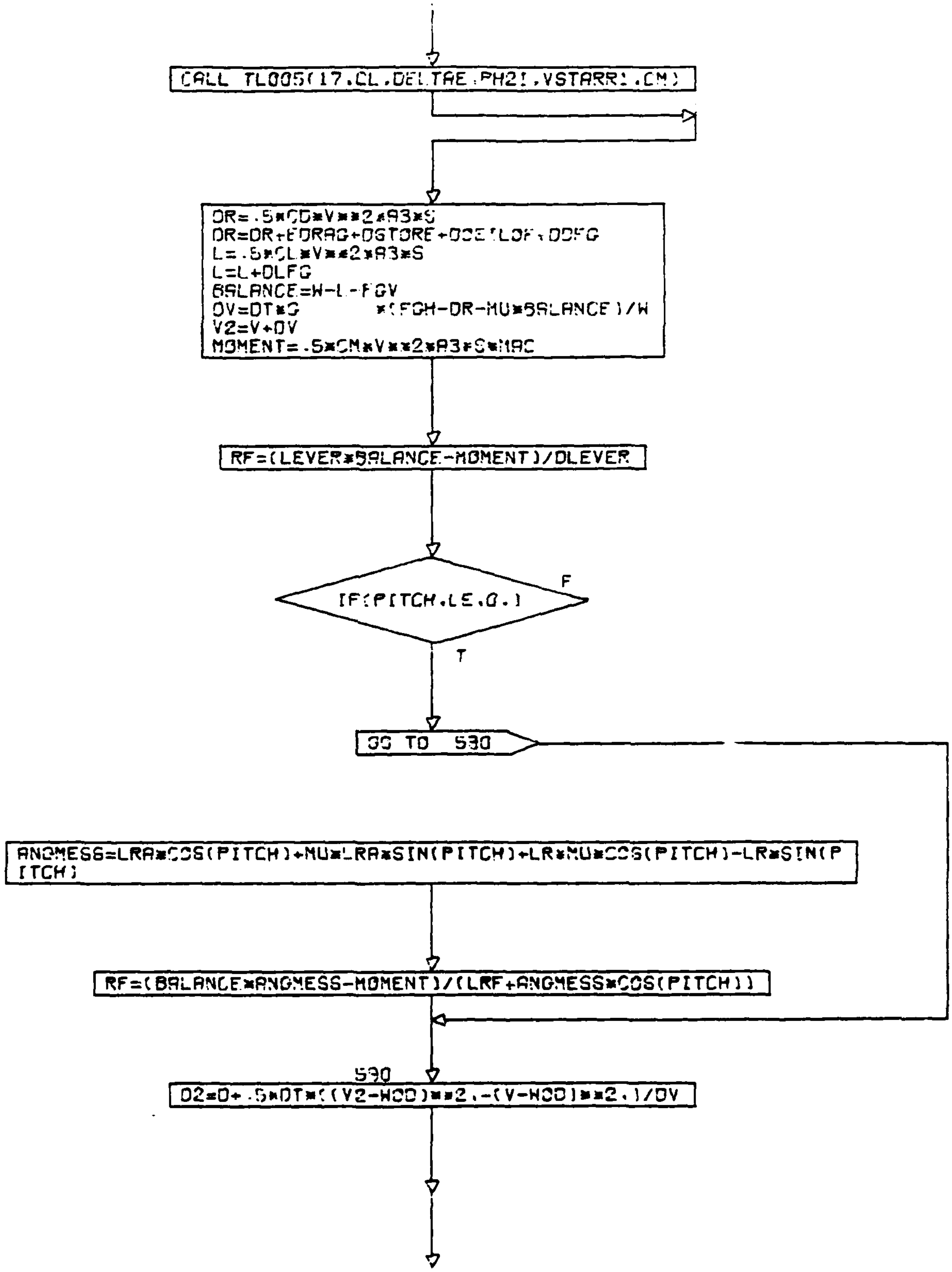
CONT. ON PG 28


IF(SEGMENT.GT.1.AND.ALPHA.GE.ALPHMIN.AND.ITRIP.EQ.0)
F
T
GO TO 550
ITRIP=0
IF(IPRINT.NE.0.AND.J.EQ.1)
F
T
PRINT 540,ALPHA,DALPHAD,IYYCG,XCG,YCG
540
FORMAT(10X,5(5X,E12.5),/,1X,8(15H AUTO ROTATION))
GO TO 260
17
63
C$$$$$$$$$$ $$$$$$$ TRY TO COUNTER ROTATION BY ELEVATOR
27
D2
550
DDELTAE=1. $ ITRIP=0

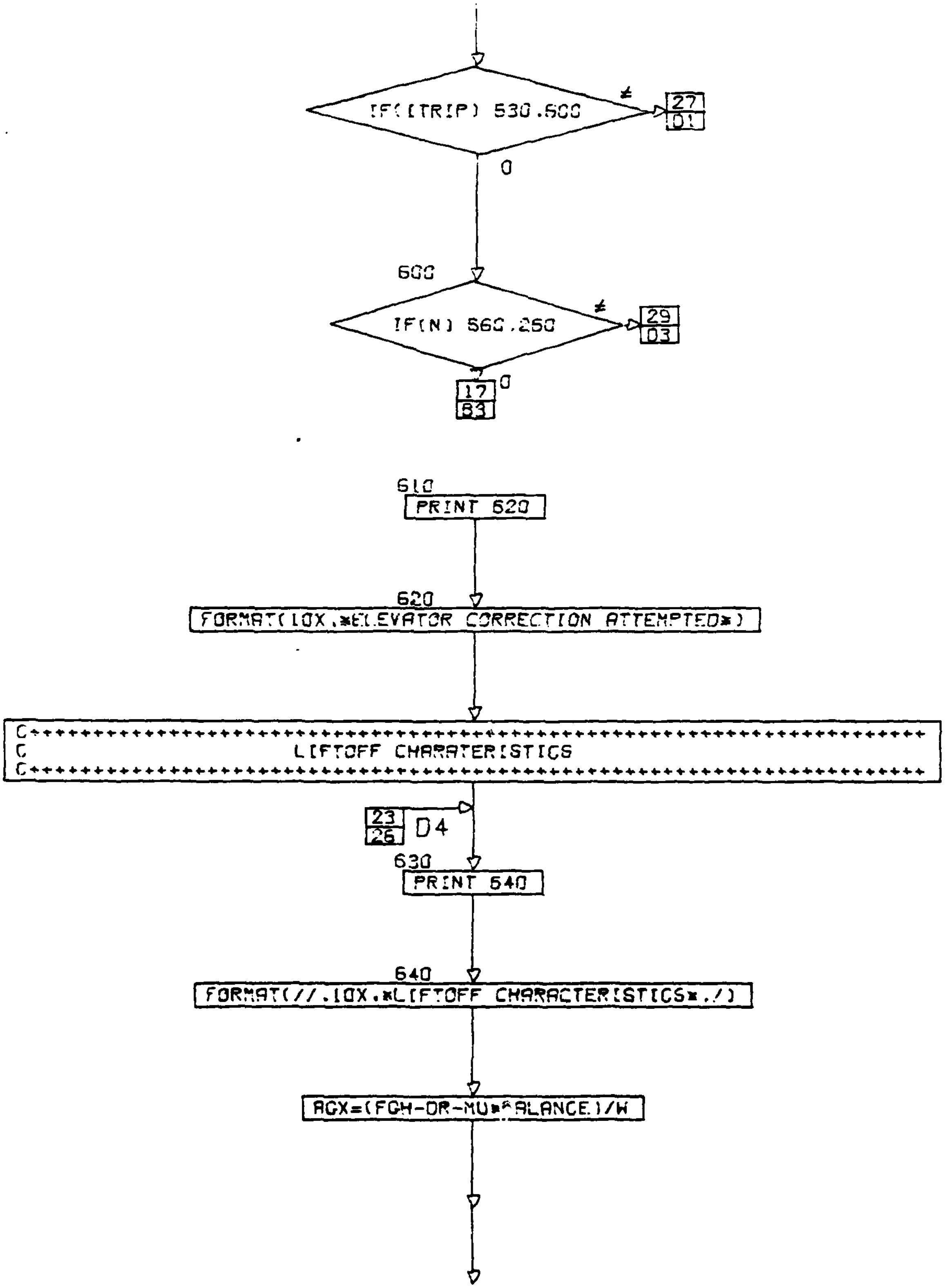
CONT. ON PG 29


D3 31
560
CALL SNEST(N,RF,DELTAE,DDELTAE,010,,ABOS,OYL,ABOYL,XCHECK,XLOC)
IF(IPRINT.NE.0.AND.J.EQ.1)
F
T
PRINT 570,N,RF,DELTAE
570
FORMAT(10X,25(1H>),5X,I10,2(10X,E12.5))
IF(DELTAE.LT.DELMAX)
F
T
GO TO 580
DELTAE=DELMAX $ N=0 $ ITRIP=1
580
CALL TLOOS(14,ALPHA,DELTAE,PH2I,VSTARRI,CL)
CALL TLOOS(13,CL,DELTAE,PH2I,VSTARRI,CD)

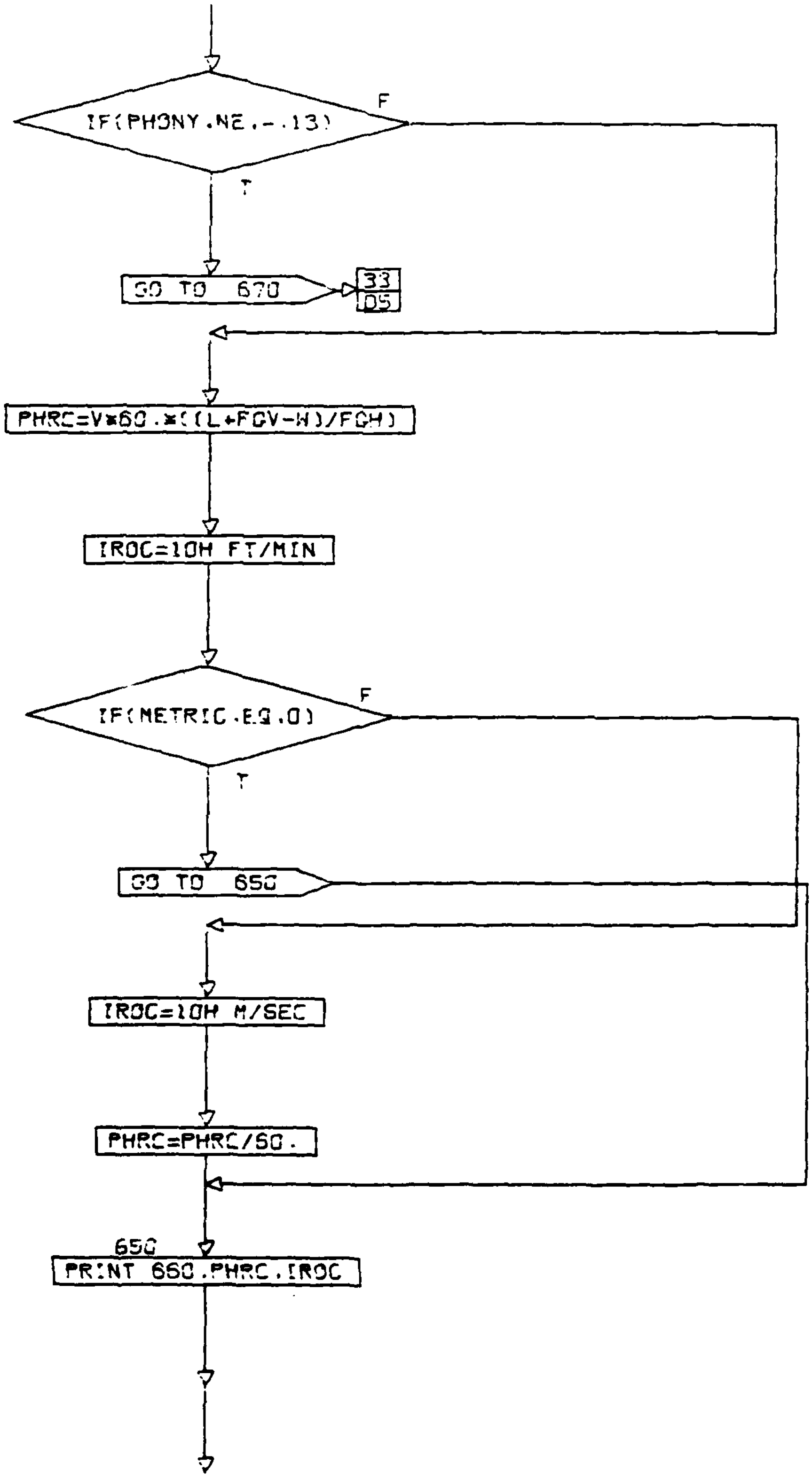
CONT. ON PG 30


CALL TL005(17.CL.DELTAE.PH2I.VSTARRI.CM)
DR=.5*CD*V**2*A3*S
DR=DR+EDRAG+DGTORE+DCE(LOF+DDFG
L=.5*CL*V**2*A3*S
L=L+DLFG
BALANCE=W-L-FGV
DV=DT*G *(FGH-DR-MU*BALANCE)/W
V2=V+DV
MOMENT=.5*CM*V**2*A3*S*MAC
RF=(LEVER*BALANCE-MOMENT)/DLEVER
IF(PITCH.LE.0.)
F
T
GO TO 590
ANGMESS=LRA*COS(PITCH)+MU*LRA*SIN(PITCH)+LR*MU*COS(PITCH)-LR*SIN(P ITCH)
RF=(BALANCE*ANGMESS-MOMENT)/(LRF+ANGMESS*COS(PITCH))
590
D2=D+.5*DT*((V2-HOD)**2.-(V-HOD)**2.)/DV

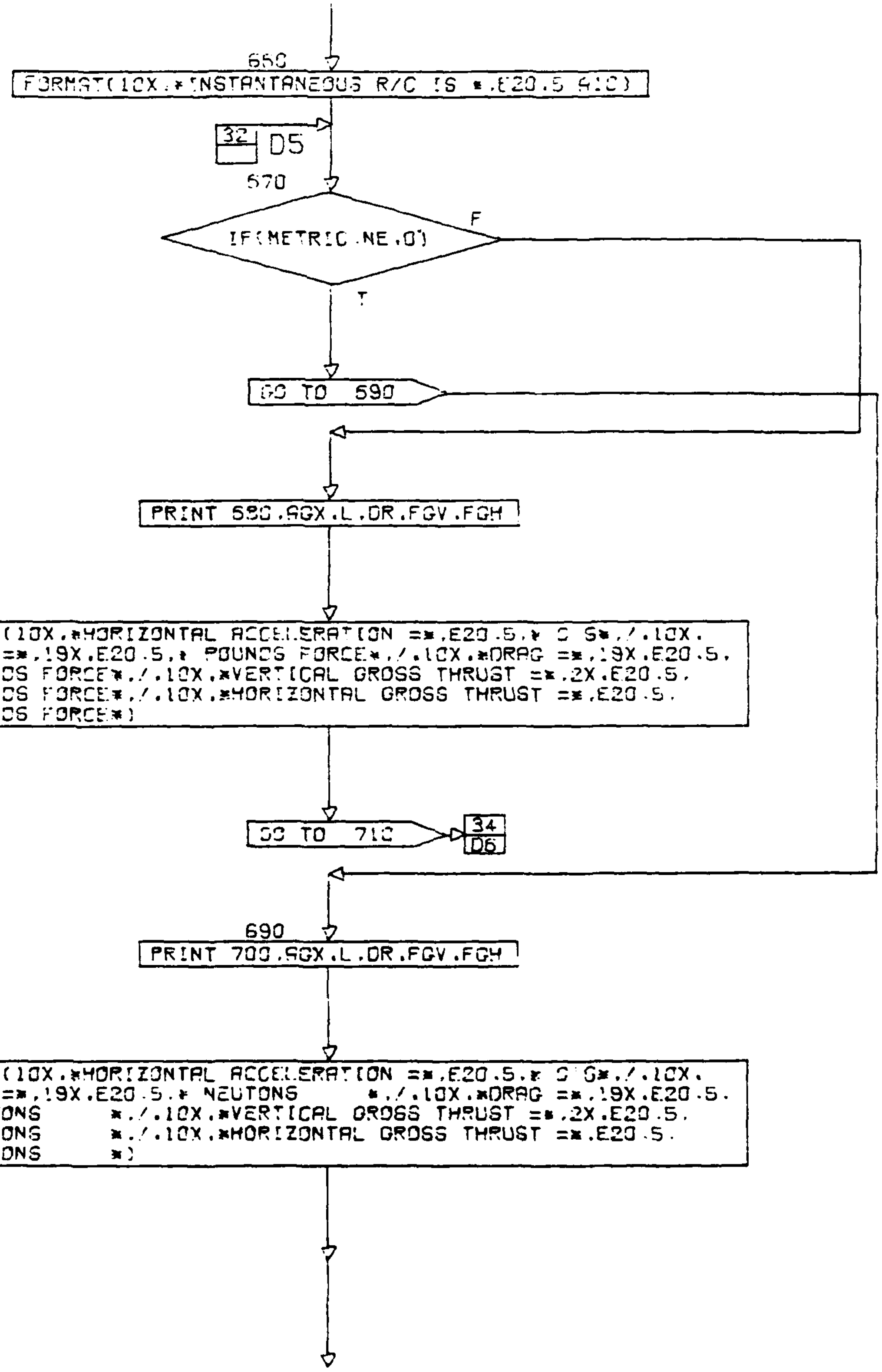
CONT. ON PG 31


IF(ITRIP) 530,600
≠
27
01
0
600
IF(N) 560,250
≠
29
03
0
17
63
610
PRINT 620
620
FORMAT(10X,*ELEVATOR CORRECTION ATTEMPTED*)
C+++
C LIFTOFF CHARATERISTICS
C+++
23
26
D4
630
PRINT 640
640
FORMAT(//,10X,*LIFTOFF CHARACTERISTICS*,/)
AGX=(FGH-DR-MU*BALANCE)/W

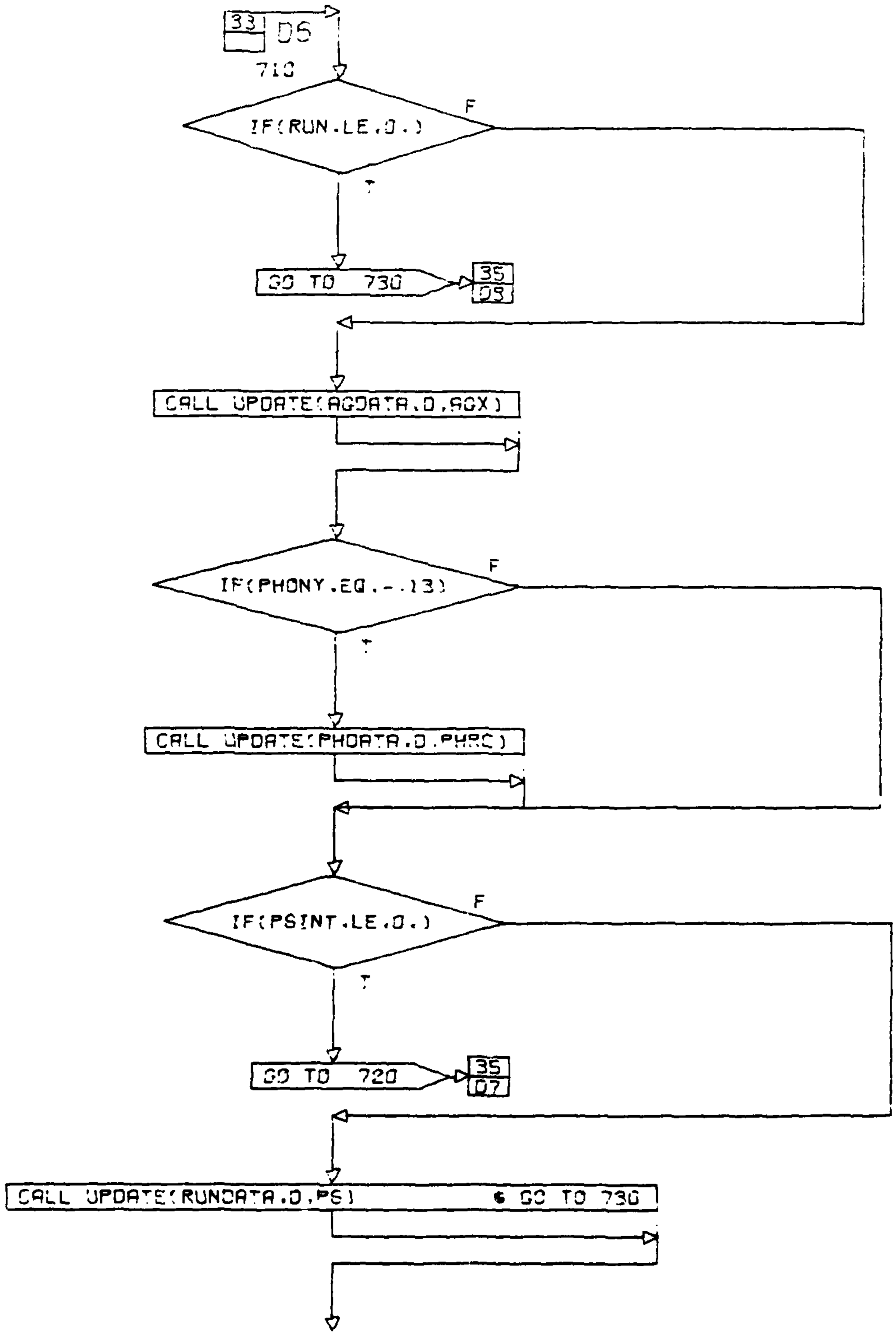
CONT. ON PG 32


IF(PHONY.NE.-.13)
F
T
GO TO 670
33
05
PHRC=V*60.*((L+FGV-W)/FGH)
IROC=10H FT/MIN
IF(METRIC.EQ.0)
F
T
GO TO 650
IROC=10H M/SEC
PHRC=PHRC/60.
650
PRINT 660,PHRC,IROC

CONT. ON PG 33


660
FORMAT(10X,*INSTANTANEOUS R/C IS *,E20.5 AIC)
32
D5
670
IF(METRIC.NE.0)
F
T
GO TO 690
PRINT 680,AGX,L,DR,FGV,FGH
FORMAT(10X,*HORIZONTAL ACCELERATION =*,E20.5,* G S*,/,10X,
LIFT =,19X,E20.5,* POUNDS FORCE*,/,10X,*DRAG =*,19X,E20.5,
* POUNDS FORCE*,/,10X,*VERTICAL GROSS THRUST =*,2X,E20.5,
* POUNDS FORCE*,/,10X,*HORIZONTAL GROSS THRUST =*,E20.5,
* POUNDS FORCE*)
GO TO 710
34
D6
690
PRINT 700,AGX,L,DR,FGV,FGH
FORMAT(10X,*HORIZONTAL ACCELERATION =*,E20.5,* G'S*,/,10X,
LIFT =,19X,E20.5,* NEUTONS *,/,10X,*DRAG =*,19X,E20.5,
* NEUTONS *,/,10X,*VERTICAL GROSS THRUST =*,2X,E20.5,
* NEUTONS *,/,10X,*HORIZONTAL GROSS THRUST =*,E20.5,
* NEUTONS *)

CONT. ON PG 34

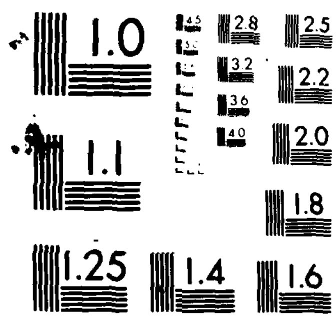

33
D6
710
IF(RUN.LE.0.)
F
T
GO TO 730
35
D5
CALL UPDATE(AGDATA.D.AGX)
IF(PHONY.EQ.-.13)
F
T
CALL UPDATE(PHDATA.D.PHRC)
IF(PSINT.LE.0.)
F
T
GO TO 720
35
D7
CALL UPDATE(RUNDATA.D.PS) $ GO TO 730

CONT. ON PG 35

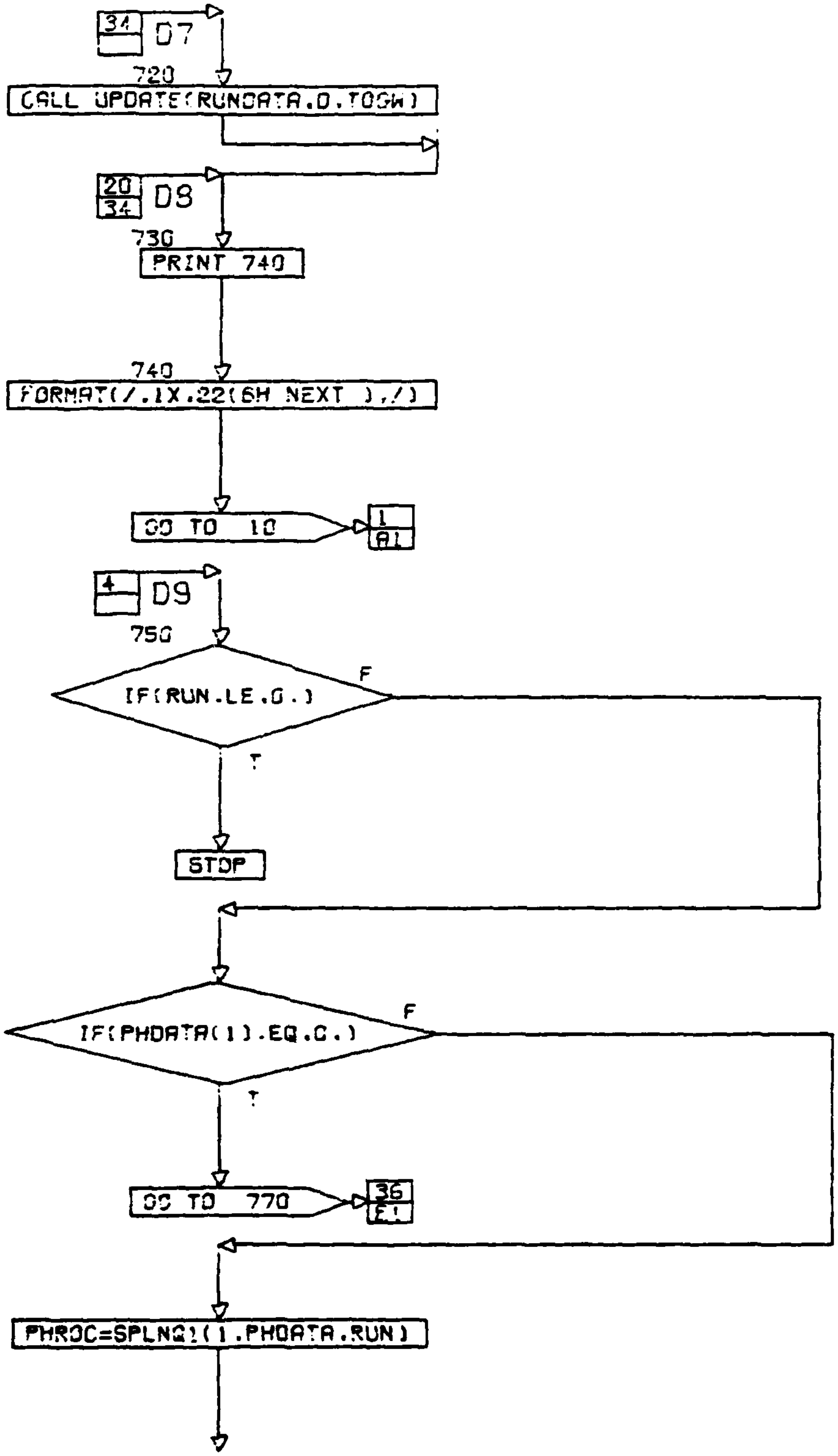
AD-A109 861   NAVAL AIR DEVELOPMENT CENTER WARMINSTER PA AIRCRAFT --ETC F/G 1/3
A SHORT TAKEOFF PERFORMANCE COMPUTER PROGRAM.(U)
NOV 81 D B KOBUS
UNCLASSIFIED NADC-81259-60 NL

2 2

END
DATE
FILMED
2 82
DTIC

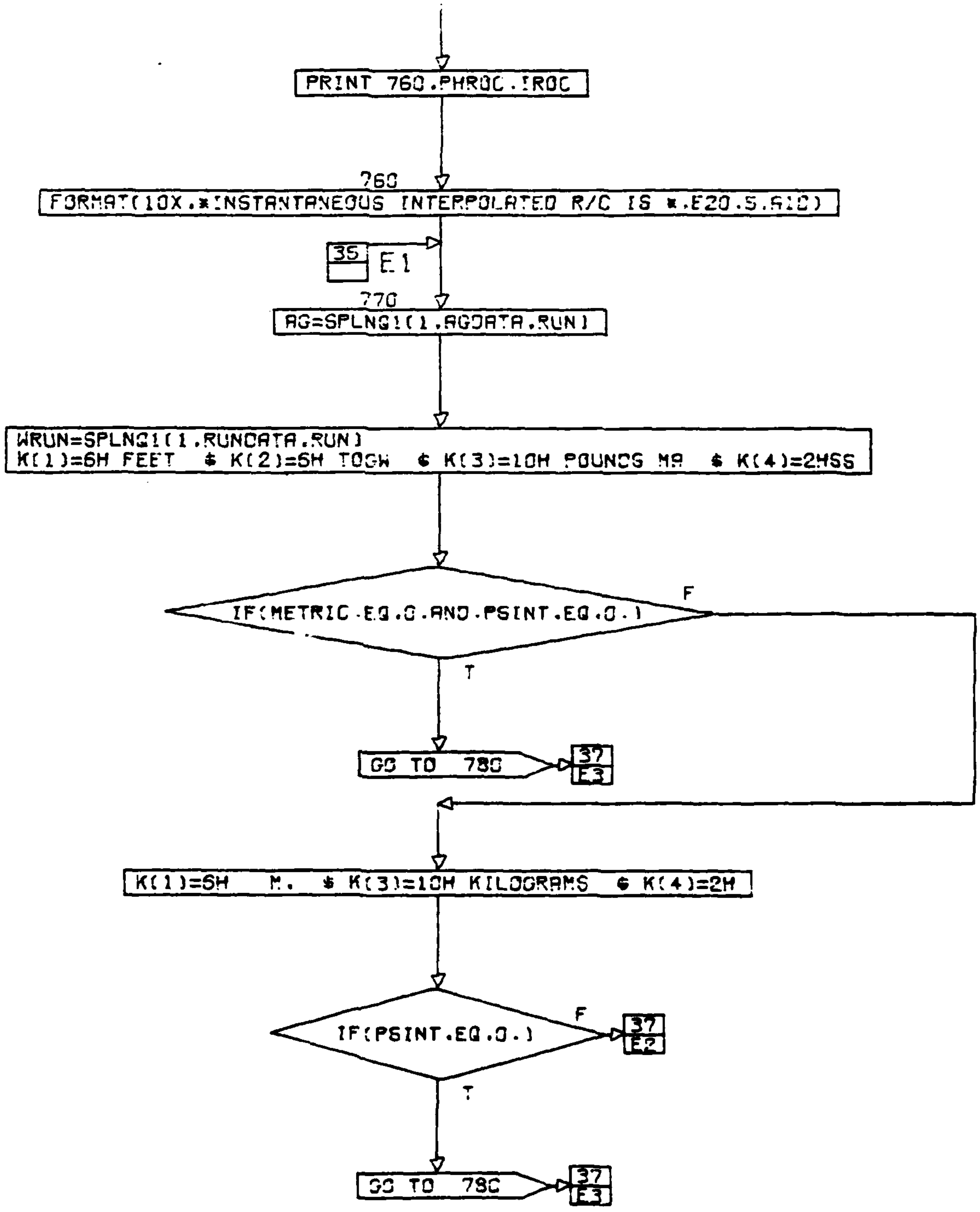


MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

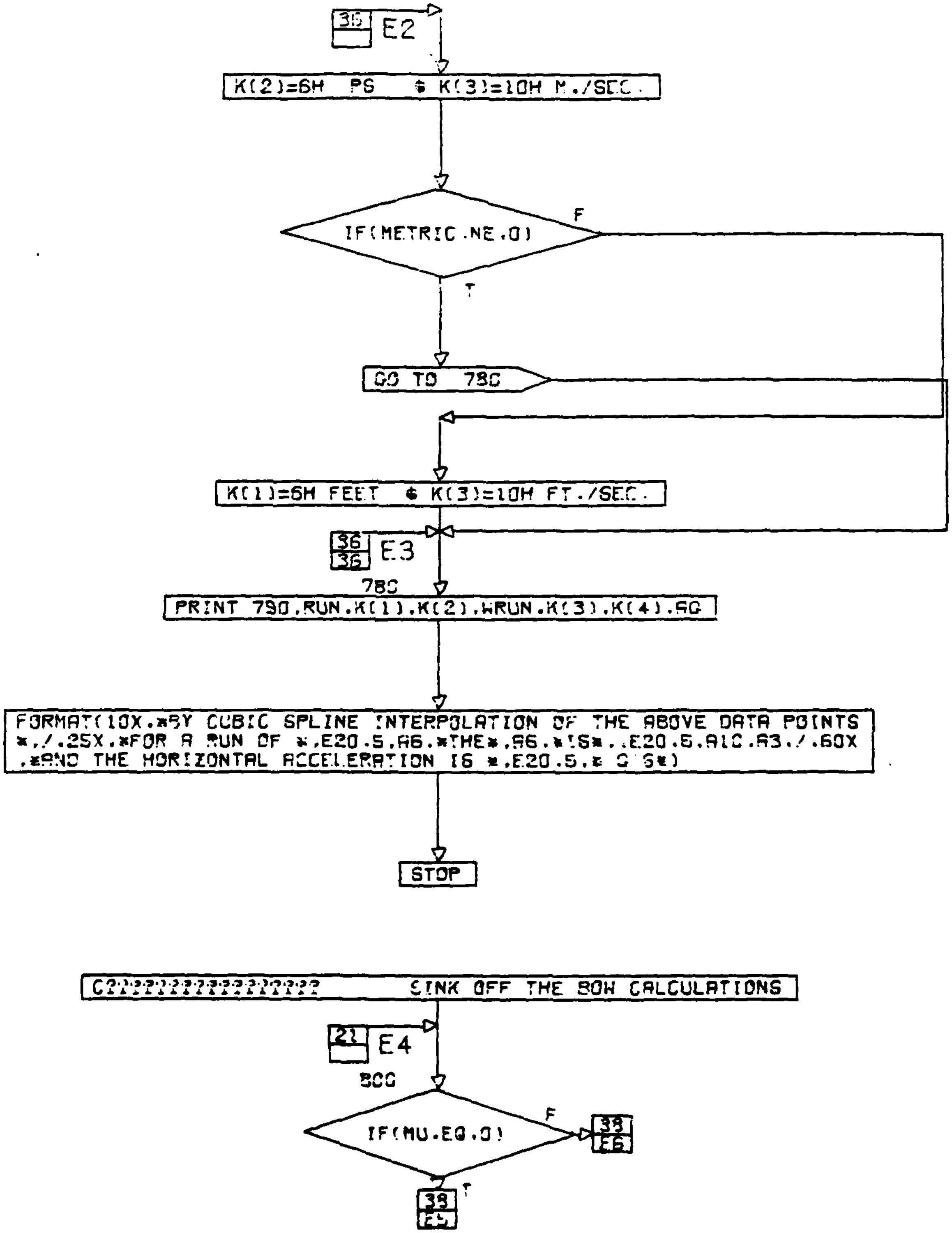


34
D7
720
CALL UPDATE(RUNDATA,0,TOGW)
20
34
D8
730
PRINT 740
740
FORMAT(/,1X,22(6H NEXT),/)
GO TO 10
1
A1
4
D9
750
IF(RUN.LE.0.)
F
T
STOP
IF(PHDATA(1).EQ.0.)
F
T
GO TO 770
36
E1
PHROC=SPLN21(1,PHDATA,RUN)

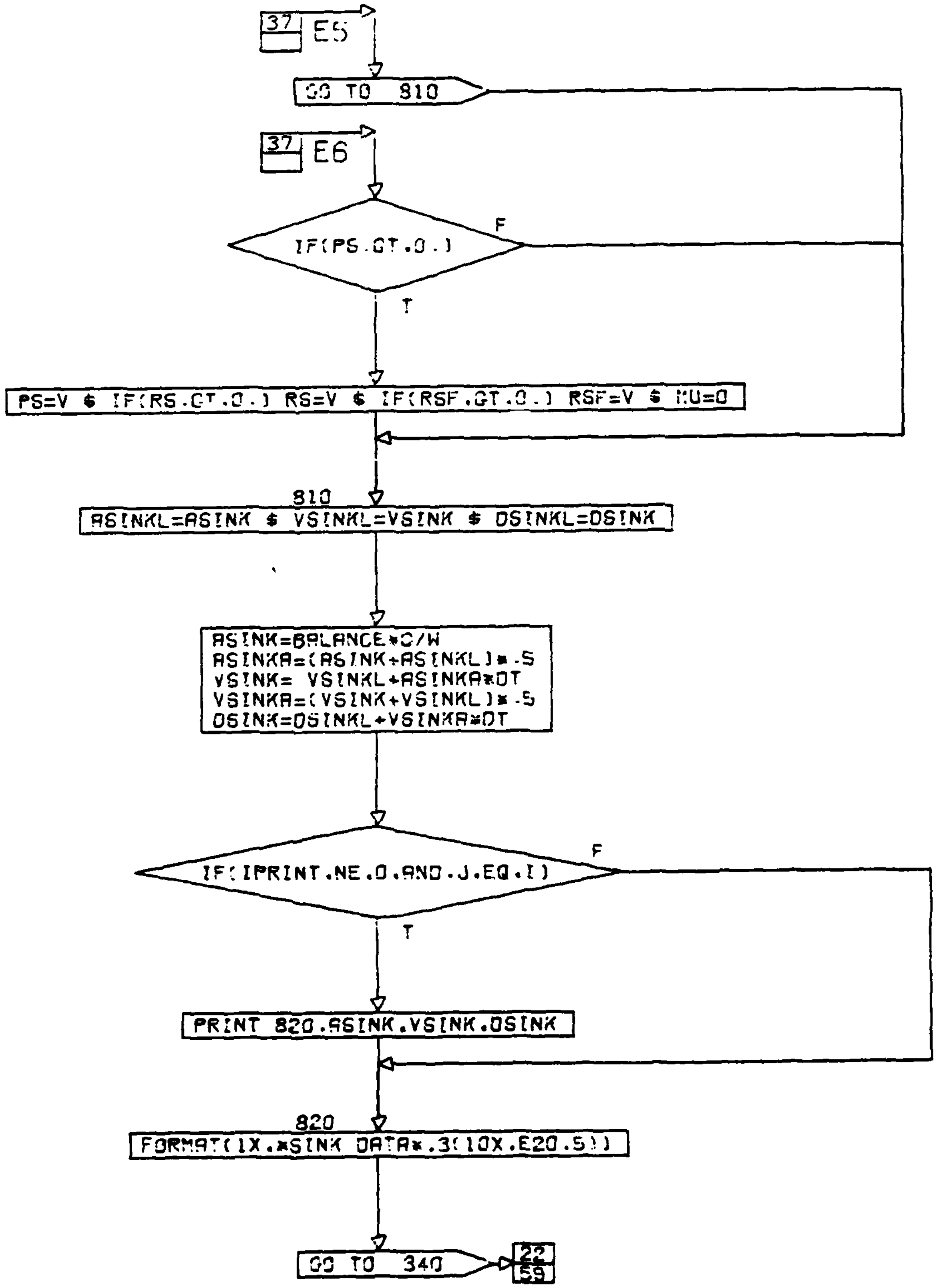
CONT. ON PG 36


PRINT 760.PHROC.IROC
760
FORMAT(10X.*INSTANTANEOUS INTERPOLATED R/C IS *.E20.5.A10)
35
E1
770
AG=SPLNG1(1.AGDATA.RUN)
WRUN=SPLNG1(1.RUNDATA.RUN)
K(1)=6H FEET $ K(2)=6H TOGW $ K(3)=10H POUNDS MA $ K(4)=2HSS
IF(METRIC.EQ.0.AND.PSINT.EQ.0.)
F
T
GO TO 780
37
E3
K(1)=6H M. $ K(3)=10H KILOGRAMS $ K(4)=2H
IF(PSINT.EQ.0.)
F
37
E2
T
GO TO 780
37
E3

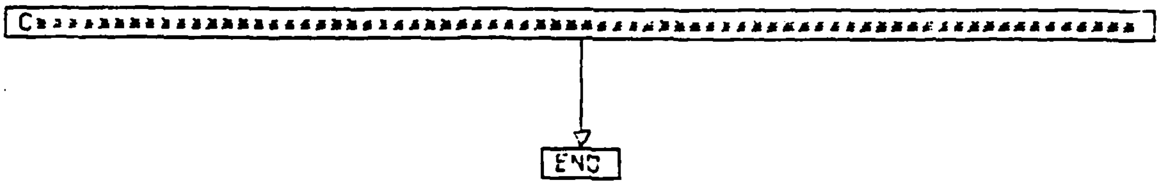
CONT. ON PG 37

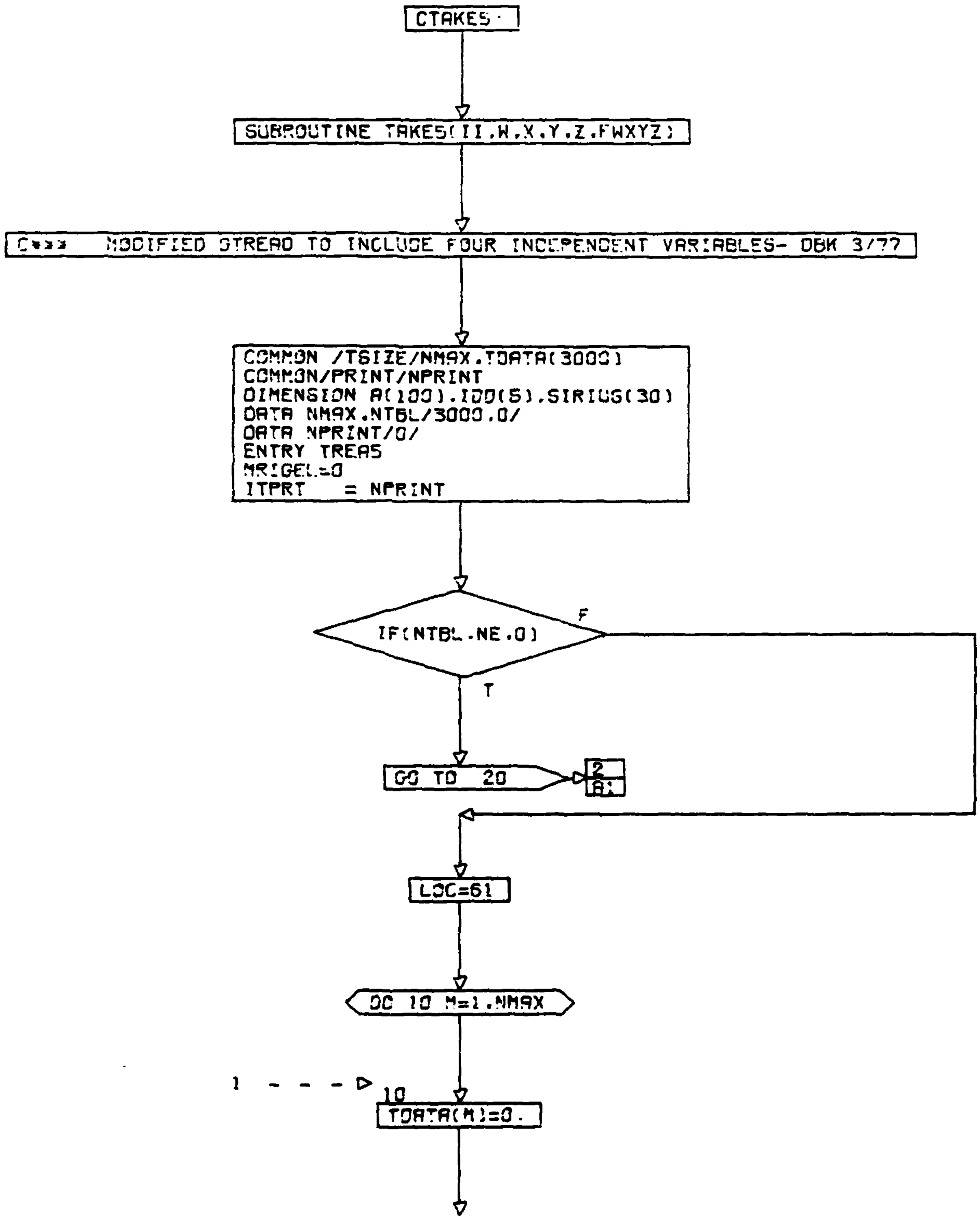

35
E2
K(2)=6H PS $ K(3)=10H M./SEC.
IF(METRIC.NE.0)
F
T
GO TO 780
K(1)=6H FEET $ K(3)=10H FT./SEC.
36
36
E3
780
PRINT 790,RUN,K(1),K(2),WRUN,K(3),K(4),AG
FORMAT(10X,*BY CUBIC SPLINE INTERPOLATION OF THE ABOVE DATA POINTS
*,/,25X,*FOR A RUN OF *,E20.5,A6,*THE*,A6,*IS*,,E20.5,A10,A3,/,60X
,*AND THE HORIZONTAL ACCELERATION IS *,E20.5,* G'S*)
STOP
C?????????????????? SINK OFF THE BOW CALCULATIONS
21
E4
800
IF(MU.EQ.0)
F
38
E6
T
38
E5

CONT. ON PG 38

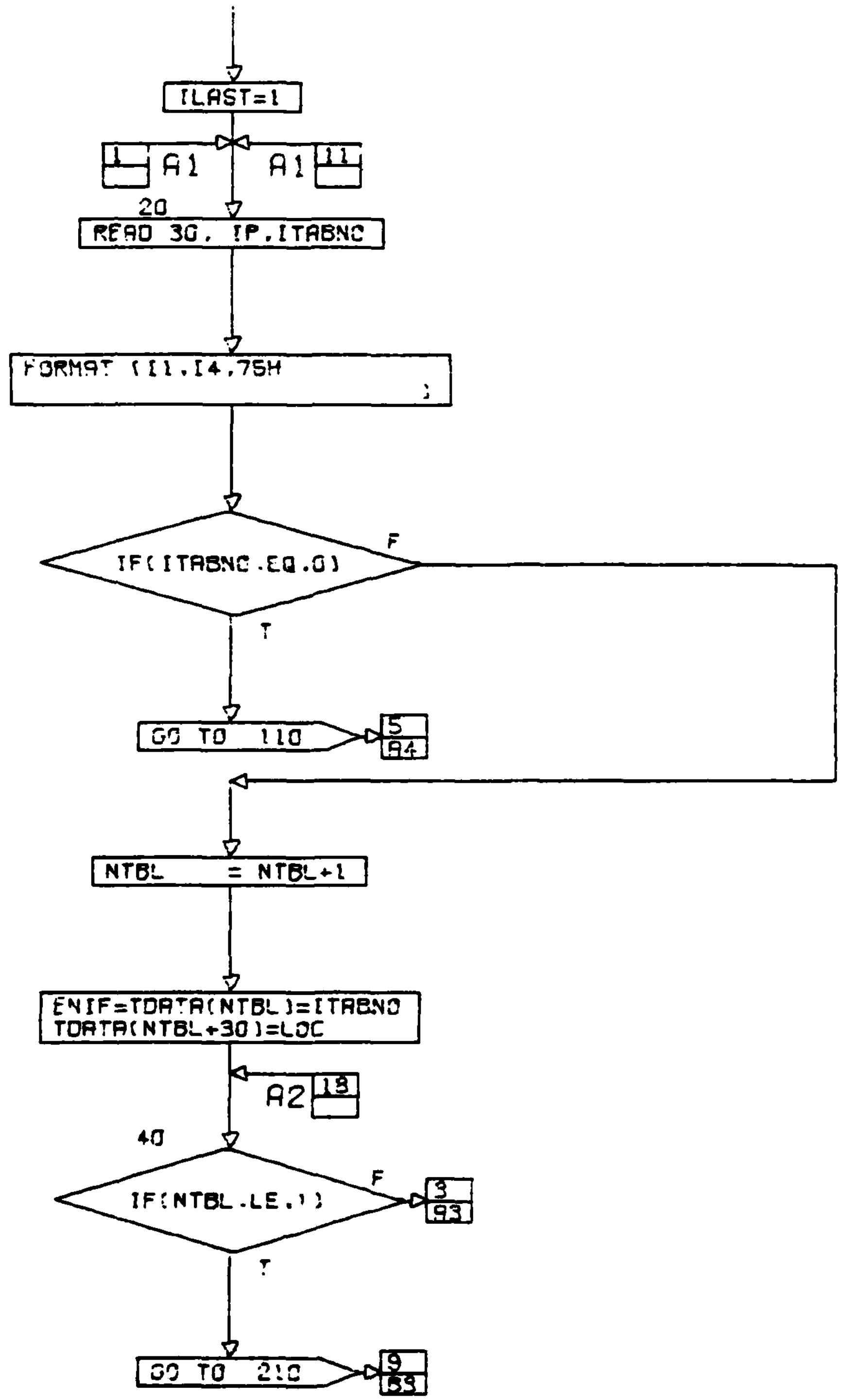

37
E5
GO TO 810
37
E6
IF(PS.GT.0.)
F
T
PS=V $ IF(RS.GT.0.) RS=V $ IF(RSF.GT.0.) RSF=V $ MU=0
810
ASINKL=ASINK $ VSINKL=VSINK $ DSINKL=DSINK
ASINK=BALANCE*G/W
ASINKA=(ASINK+ASINKL)*.5
VSINK= VSINKL+ASINKA*DT
VSINKA=(VSINK+VSINKL)*.5
DSINK=DSINKL+VSINKA*DT
IF(IPRINT.NE.0.AND.J.EQ.I)
F
T
PRINT 820,ASINK,VSINK,DSINK
820
FORMAT(1X,*SINK DATA*,3(10X,E20.5))
GO TO 340
22
59

CONT. ON PG 39

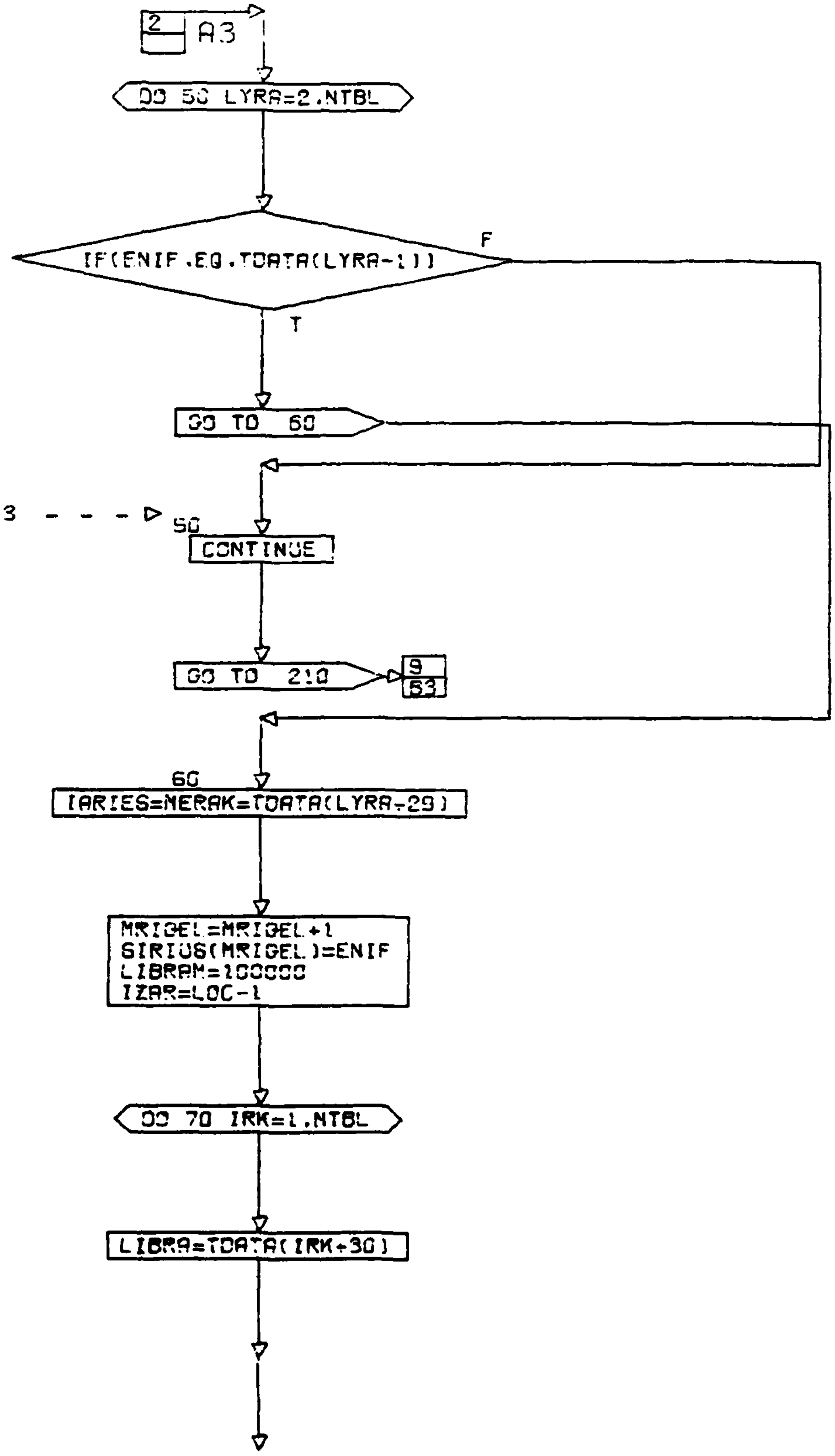

END


CTAKES
SUBROUTINE TAKES(II,W,X,Y,Z,FWXYZ)
C*** MODIFIED STREAD TO INCLUDE FOUR INDEPENDENT VARIABLES- DBK 3/77
COMMON /TSIZE/NMAX,TDATA(3000)
COMMON/PRINT/NPRINT
DIMENSION A(100),IDD(5),SIRIUS(30)
DATA NMAX,NTBL/3000,0/
DATA NPRINT/0/
ENTRY TREAS
MRIGEL=0
ITPRT = NPRINT
IF(NTBL.NE.0)
F
T
GO TO 20
2
A1
LOC=61
DO 10 M=1,NMAX
1
10
TDATA(M)=0.

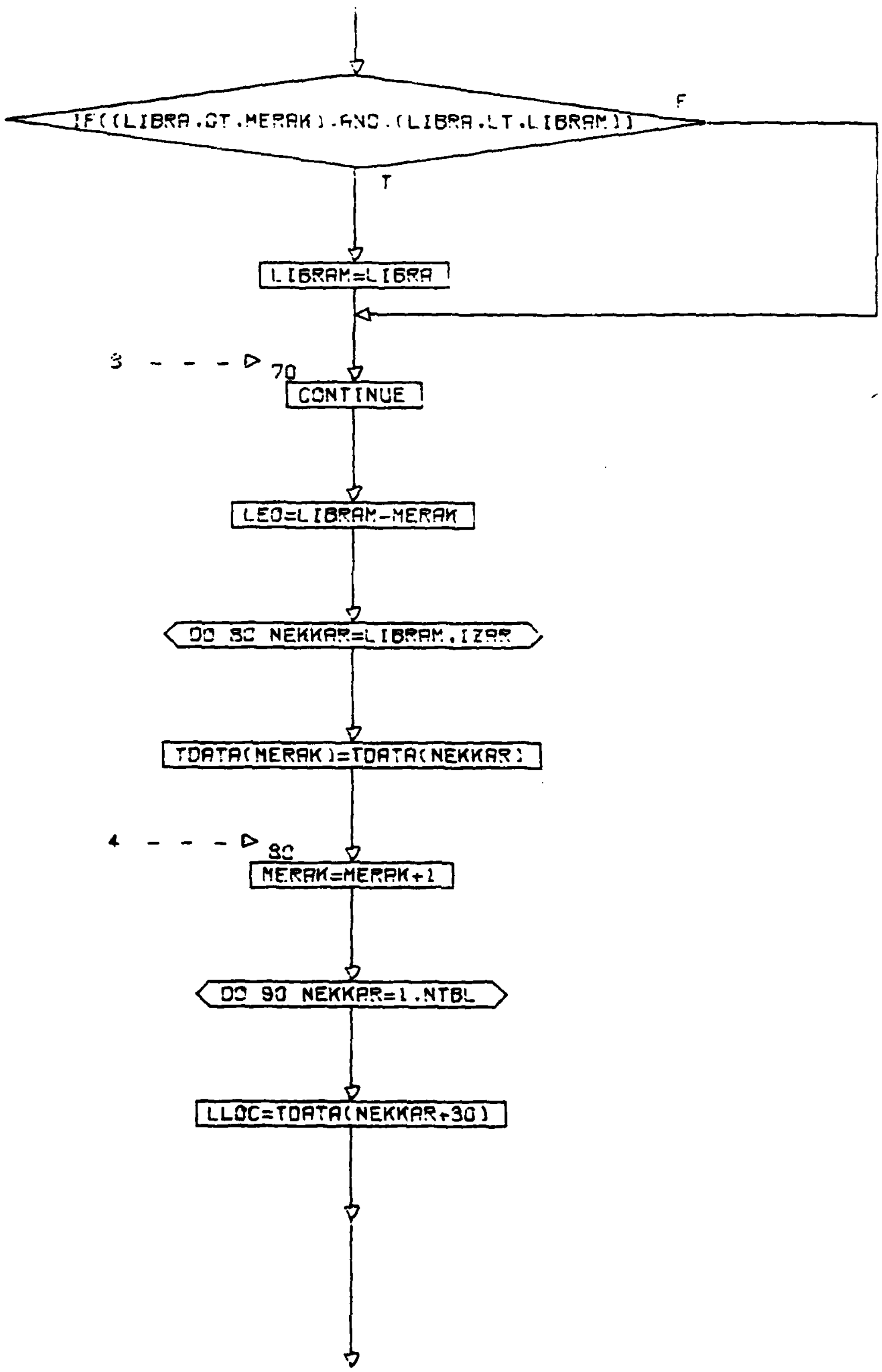
CONT. ON PG 2


ILAST=1
1
A1
A1
11
20
READ 30, IP,ITABNO
FORMAT (I1,I4,75H
)
IF(ITABNO.EQ.0)
F
T
GO TO 110
5
A4
NTBL = NTBL+1
ENIF=TDATA(NTBL)=ITABNO
TDATA(NTBL+30)=LOC
A2
18
40
IF(NTBL.LE.1)
F
3
A3
T
GO TO 210
9
B3

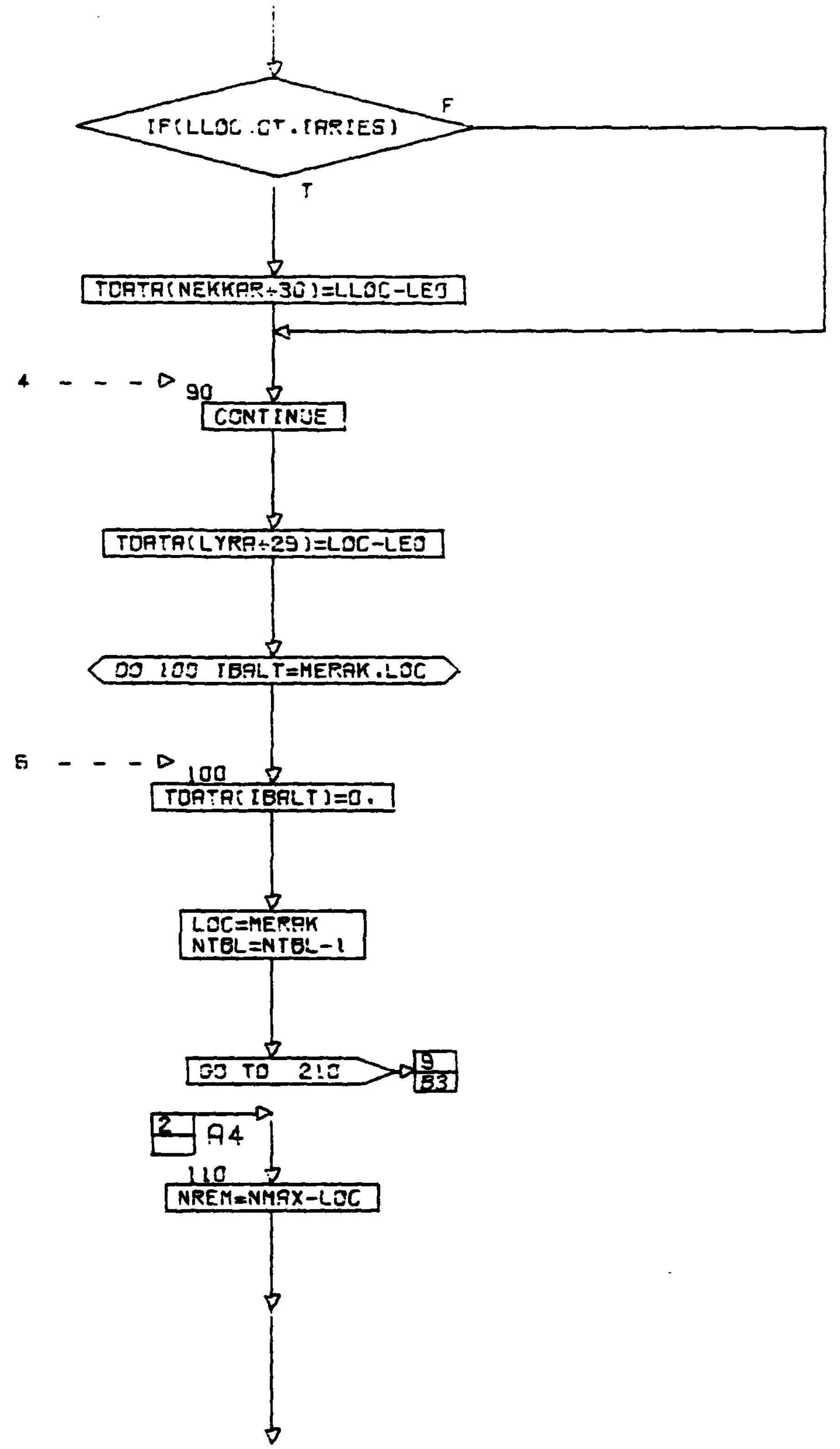
CONT. ON PG 3


2
A3
DO 50 LYRA=2,NTBL
IF(ENIF.EQ.TDATA(LYRA-1))
F
T
GO TO 60
3
50
CONTINUE
GO TO 210
9
53
60
IARIES=NERAK=TDATA(LYRA+29)
MRIGEL=MRIGEL+1
SIRIUS(MRIGEL)=ENIF
LIBRAM=100000
IZAR=LOC-1
DO 70 IRK=1,NTBL
LIBRA=TDATA(IRK+30)

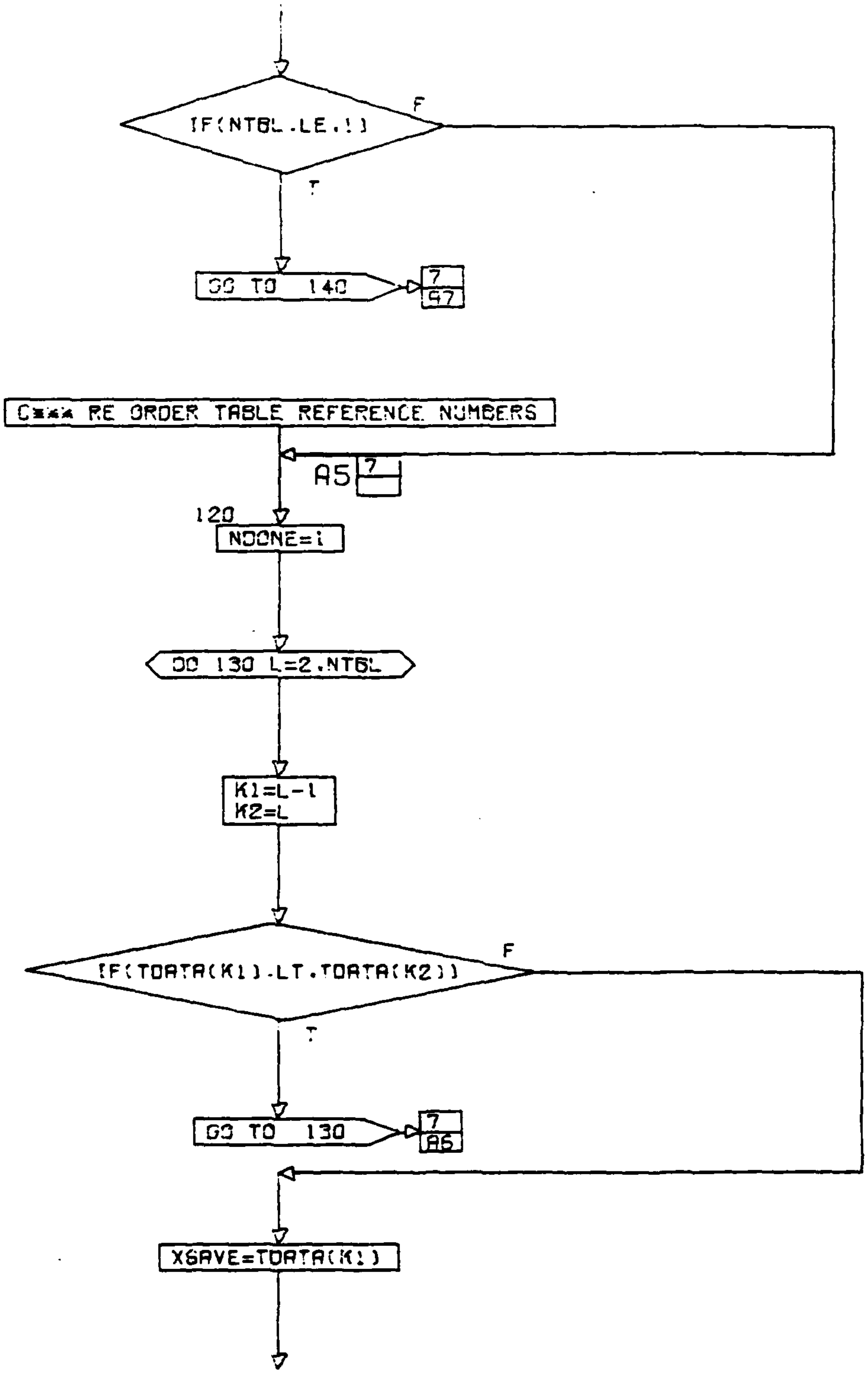
CONT. ON PG 4


IF((LIBRA.GT.MERAK).AND.(LIBRA.LT.LIBRAM))
F
T
LIBRAM=LIBRA
3
70
CONTINUE
LEO=LIBRAM-MERAK
DO 80 NEKKAR=LIBRAM,IZAR
TDATA(MERAK)=TDATA(NEKKAR)
4
80
MERAK=MERAK+1
DO 90 NEKKAR=1,NTBL
LLOC=TDATA(NEKKAR+30)

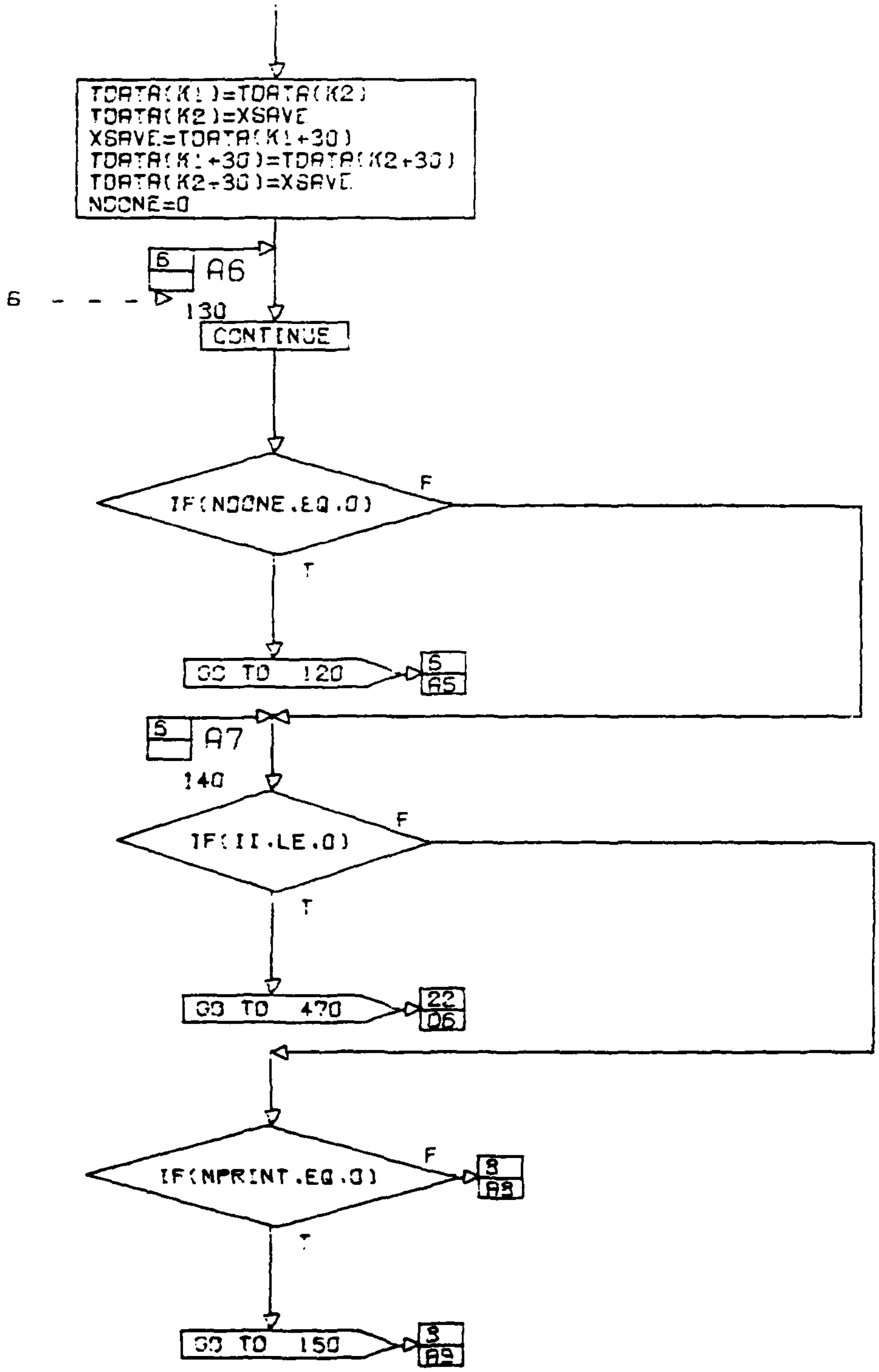
CONT. ON PG 5


IF(LLOC.GT.IARIES)
F
T
TDATA(NEKKAR+30)=LLOC-LEO
4
90
CONTINUE
TDATA(LYRA+29)=LOC-LEO
DO 100 IBALT=MERAK,LOC
6
100
TDATA(IBALT)=0.
LOC=MERAK
NTBL=NTBL-1
GO TO 210
9
B3
2
A4
110
NREM=NMAX-LOC

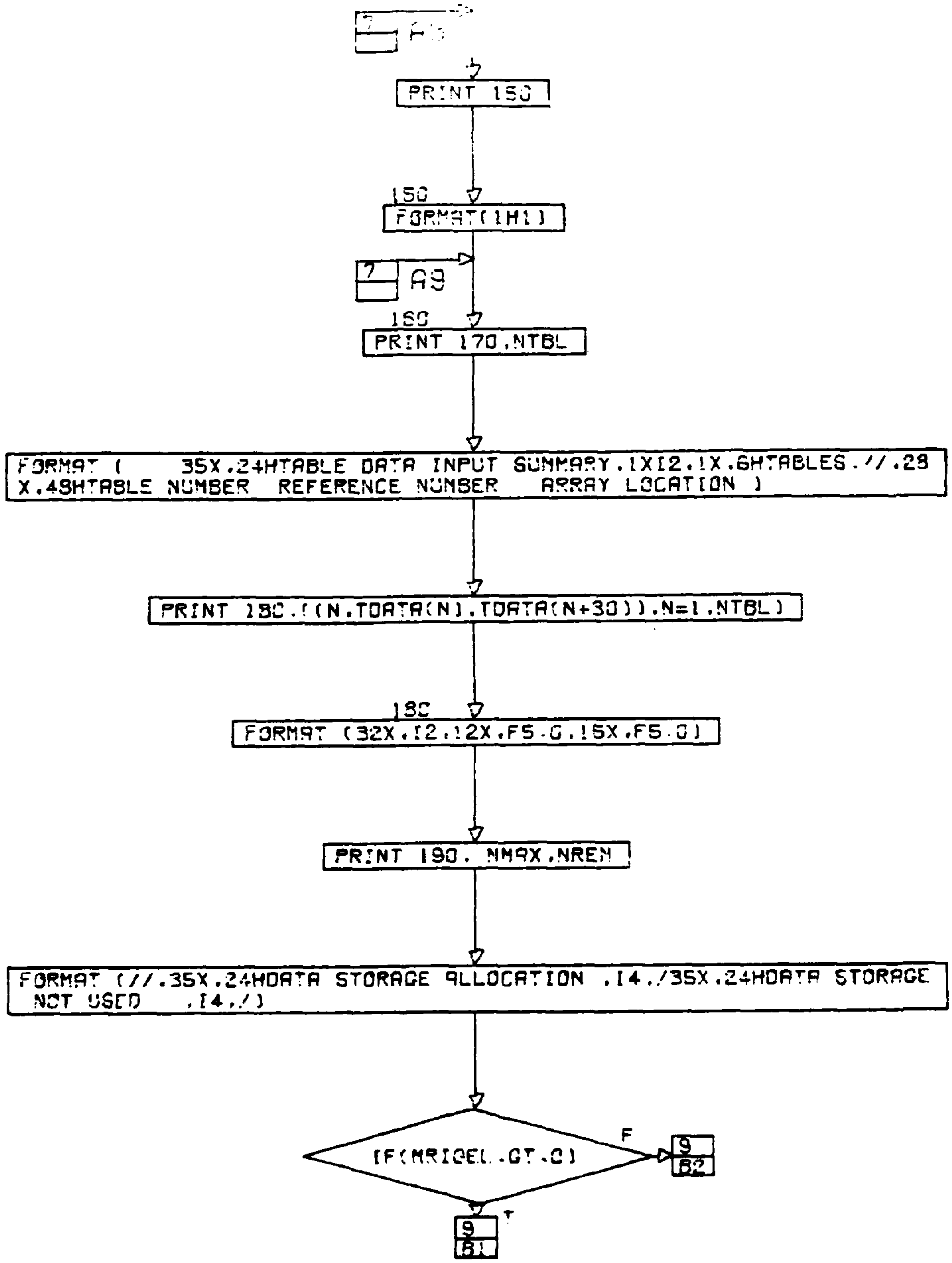
CONT. ON PG 6


IF(NTBL.LE.1)
F
T
GO TO 140
7
97
C*** RE ORDER TABLE REFERENCE NUMBERS
A5
7
120
NDONE=1
DO 130 L=2,NTBL
K1=L-1
K2=L
IF(TDATA(K1).LT.TDATA(K2))
F
T
GO TO 130
7
A6
XSAVE=TDATA(K1)

CONT. ON PG 7

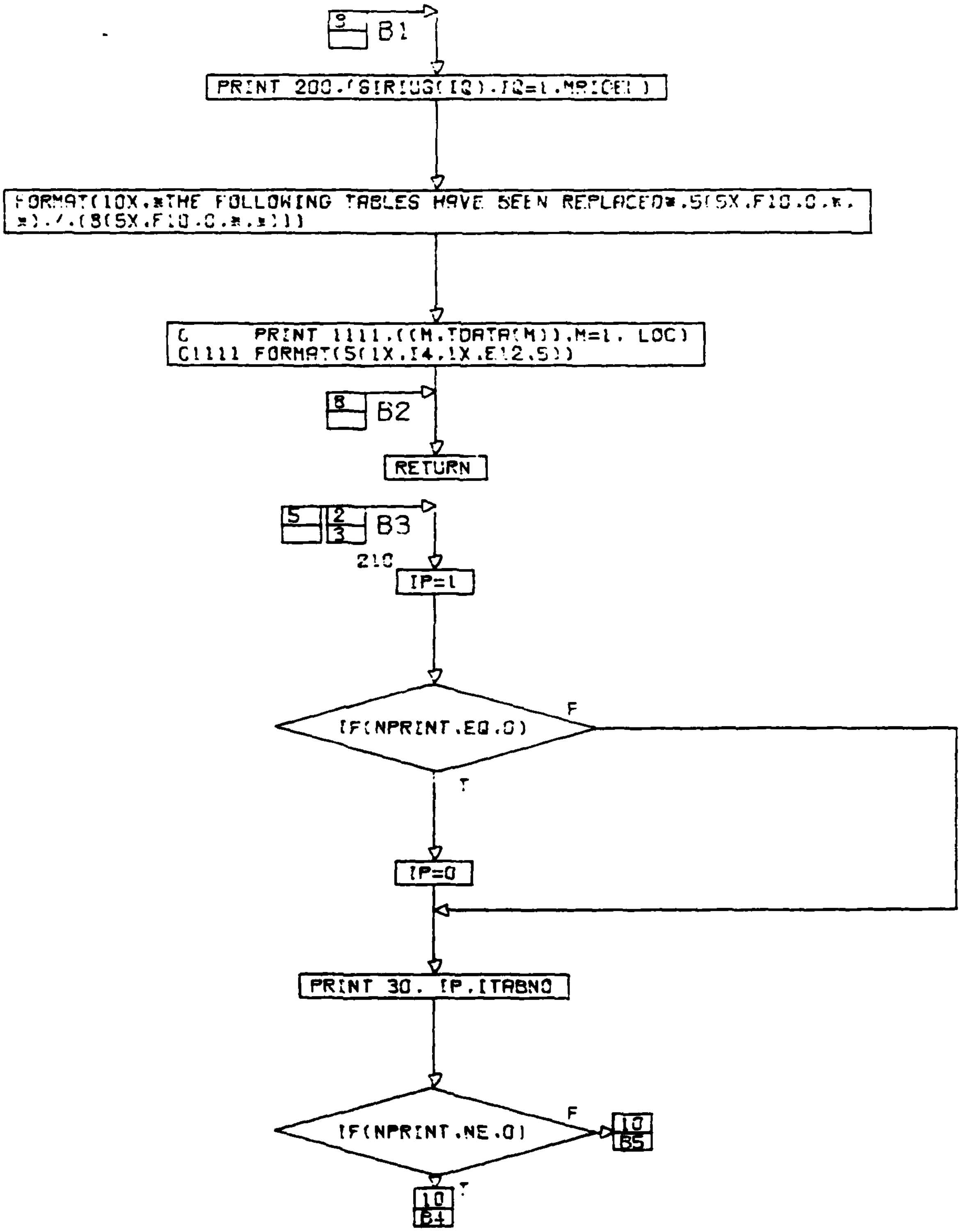

TDATA(K1)=TDATA(K2)
TDATA(K2)=XSAVE
XSAVE=TDATA(K1+30)
TDATA(K1+30)=TDATA(K2+30)
TDATA(K2+30)=XSAVE
NDONE=0
6
A6
6
130
CONTINUE
IF(NDONE.EQ.0)
F
T
GO TO 120
6
A5
5
A7
140
IF(II.LE.0)
F
T
GO TO 470
22
D6
IF(NPRINT.EQ.0)
F
8
A8
T
GO TO 150
8
A9

CONT. ON PG 8

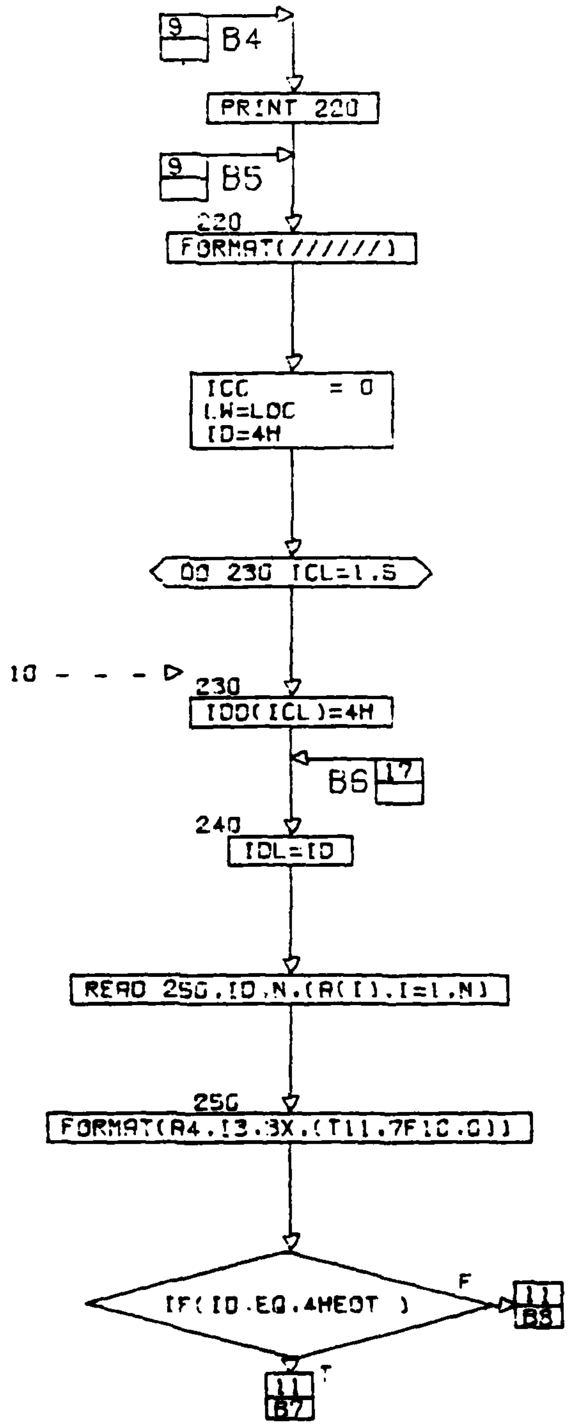


7
A9
PRINT 150
150
FORMAT(1H1)
7
A9
160
PRINT 170,NTBL
FORMAT (35X,24HTABLE DATA INPUT SUMMARY,1XI2,1X,6HTABLES,//,28
X,48HTABLE NUMBER REFERENCE NUMBER ARRAY LOCATION)
PRINT 180,((N,TDATA(N),TDATA(N+30)),N=1,NTBL)
180
FORMAT (32X,I2,12X,F5.0,15X,F5.0)
PRINT 190, NMAX,NREM
FORMAT (//,35X,24HDATA STORAGE ALLOCATION ,I4,/35X,24HDATA STORAGE
NOT USED ,I4,/)
IF(MRIGEL.GT.0)
F
9
B2
T
9
B1

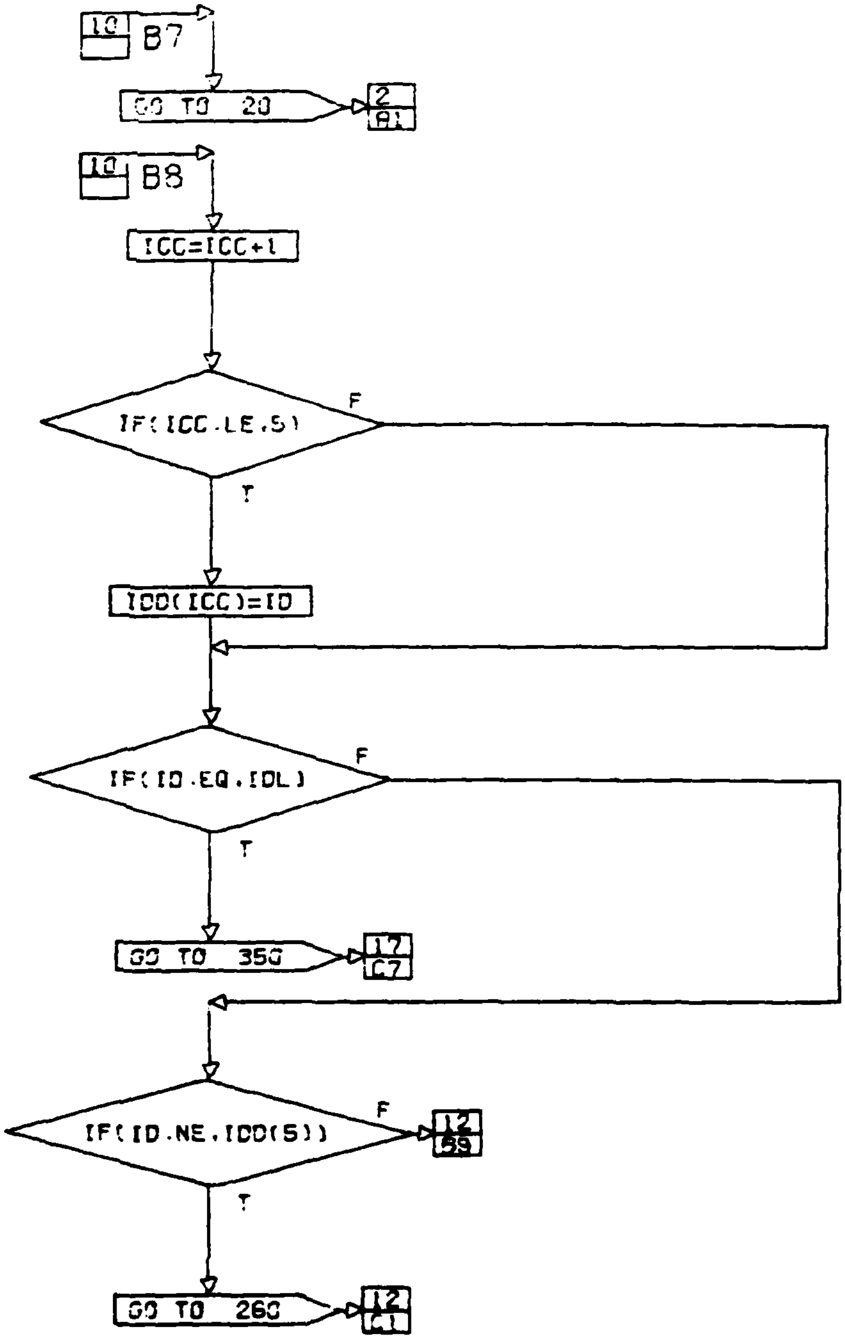
CONT. ON PG 9


9
B1
PRINT 200.(SIRIUS(IQ).IQ=1.MRICE1)
FORMAT(10X.*THE FOLLOWING TABLES HAVE BEEN REPLACED*.5(5X.F10.0.*.
)./.(8(5X.F10.0..*)))
C PRINT 1111.((M.TDATA(M)).M=1. LOC)
C1111 FORMAT(5(1X.I4.1X.E12.5))
8
B2
RETURN
5
2
3
B3
210
IP=1
IF(NPRINT.EQ.0)
F
T
IP=0
PRINT 30. IP.ITABNO
IF(NPRINT.NE.0)
F
10
B5
T
10
B4

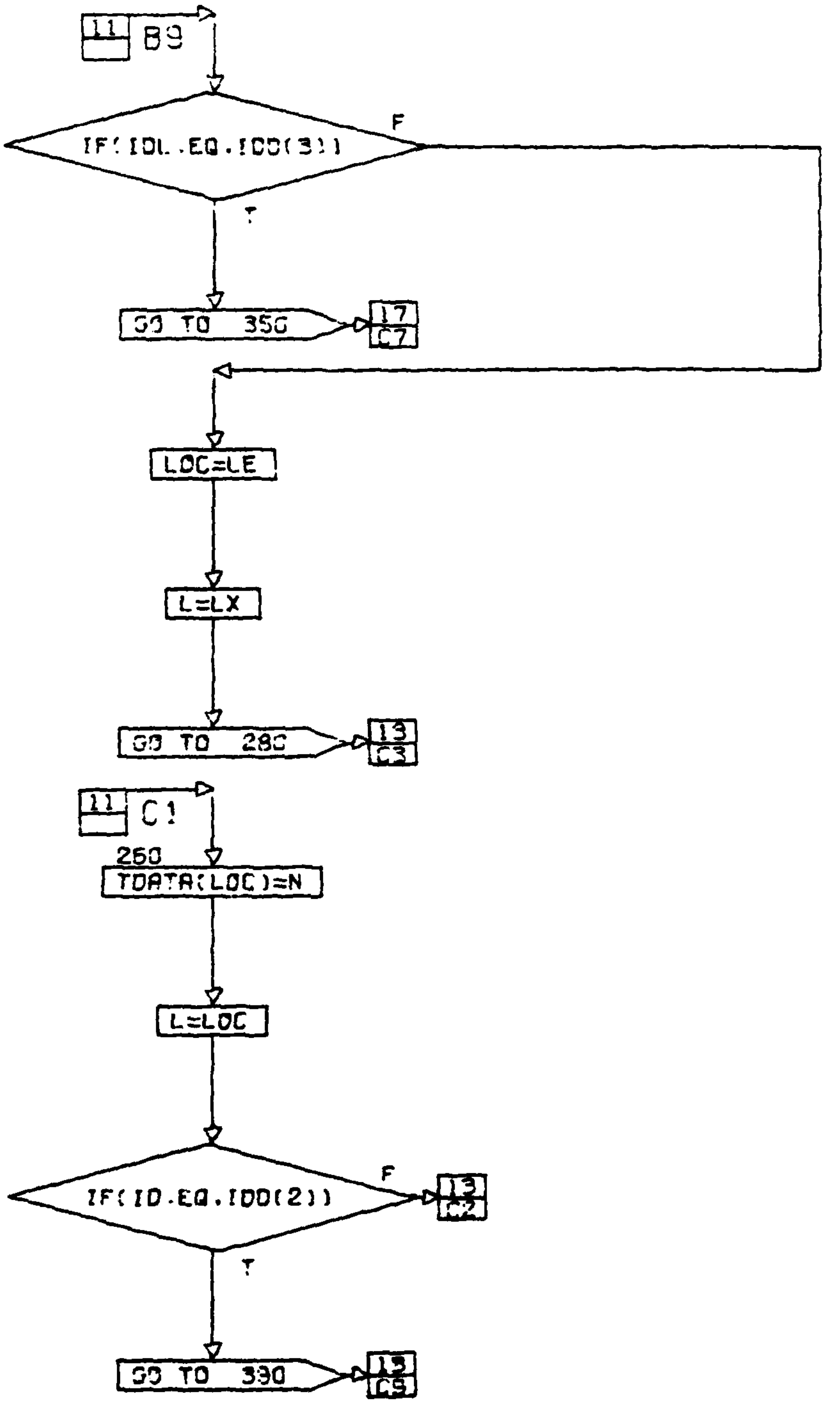
CONT. ON PG 10


9
B4
PRINT 220
9
B5
220
FORMAT(//////)
ICC = 0
LW=LOC
ID=4H
DO 230 ICL=1,5
10
230
IDD(ICL)=4H
B6
17
240
IDL=ID
READ 250,ID,N,(A(I),I=1,N)
250
FORMAT(A4,I3,3X,(T11,7F10.0))
IF(ID.EQ.4HEOT)
F
11
B8
T
11
B7

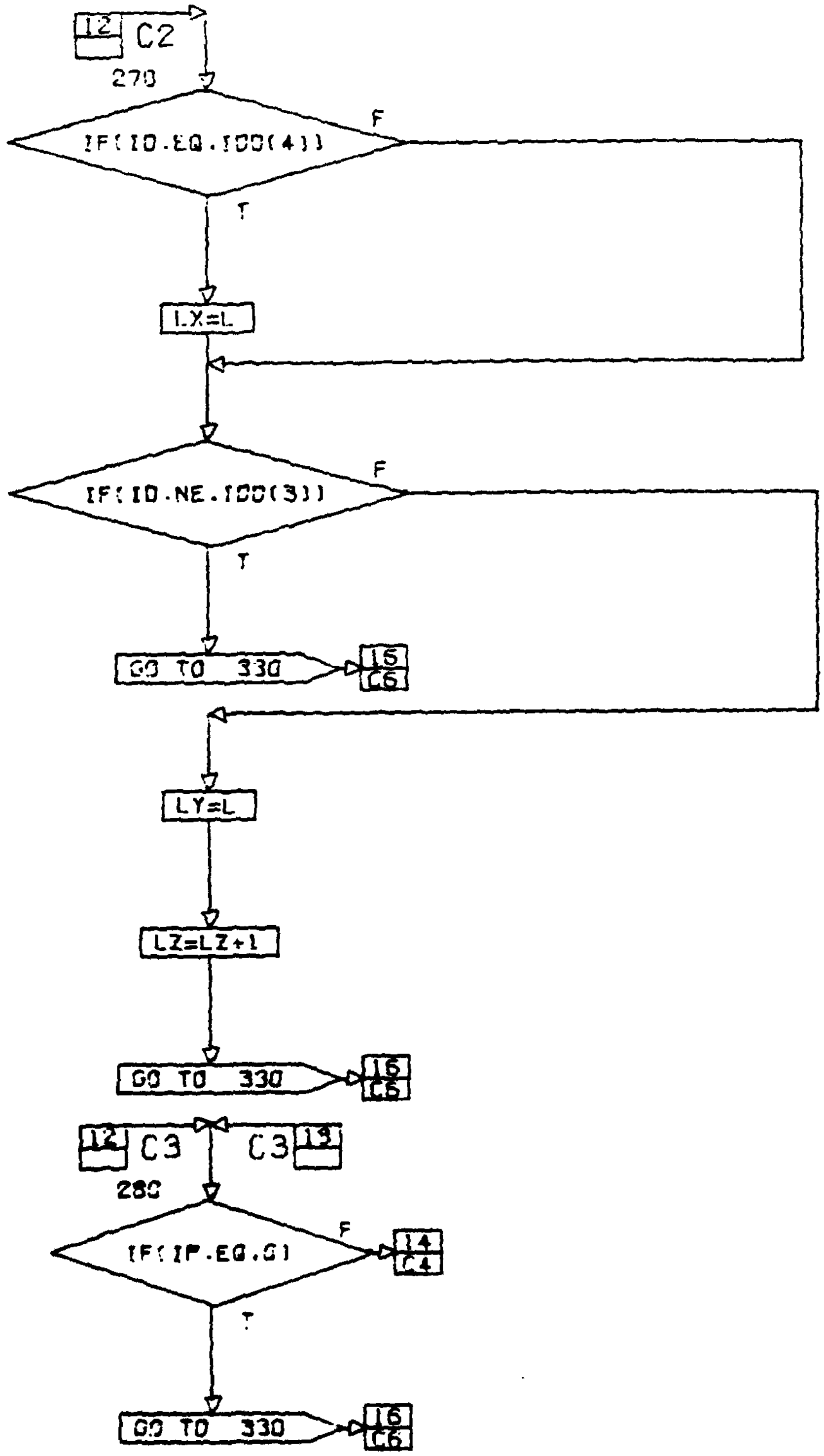
CONT. ON PG 11


10
B7
GO TO 20
2
A1
10
B8
ICC=ICC+1
IF(ICC.LE.5)
F
T
IDD(ICC)=ID
IF(ID.EQ.IDL)
F
T
GO TO 350
17
C7
IF(ID.NE.IDD(5))
F
12
B9
T
GO TO 260
12
C1

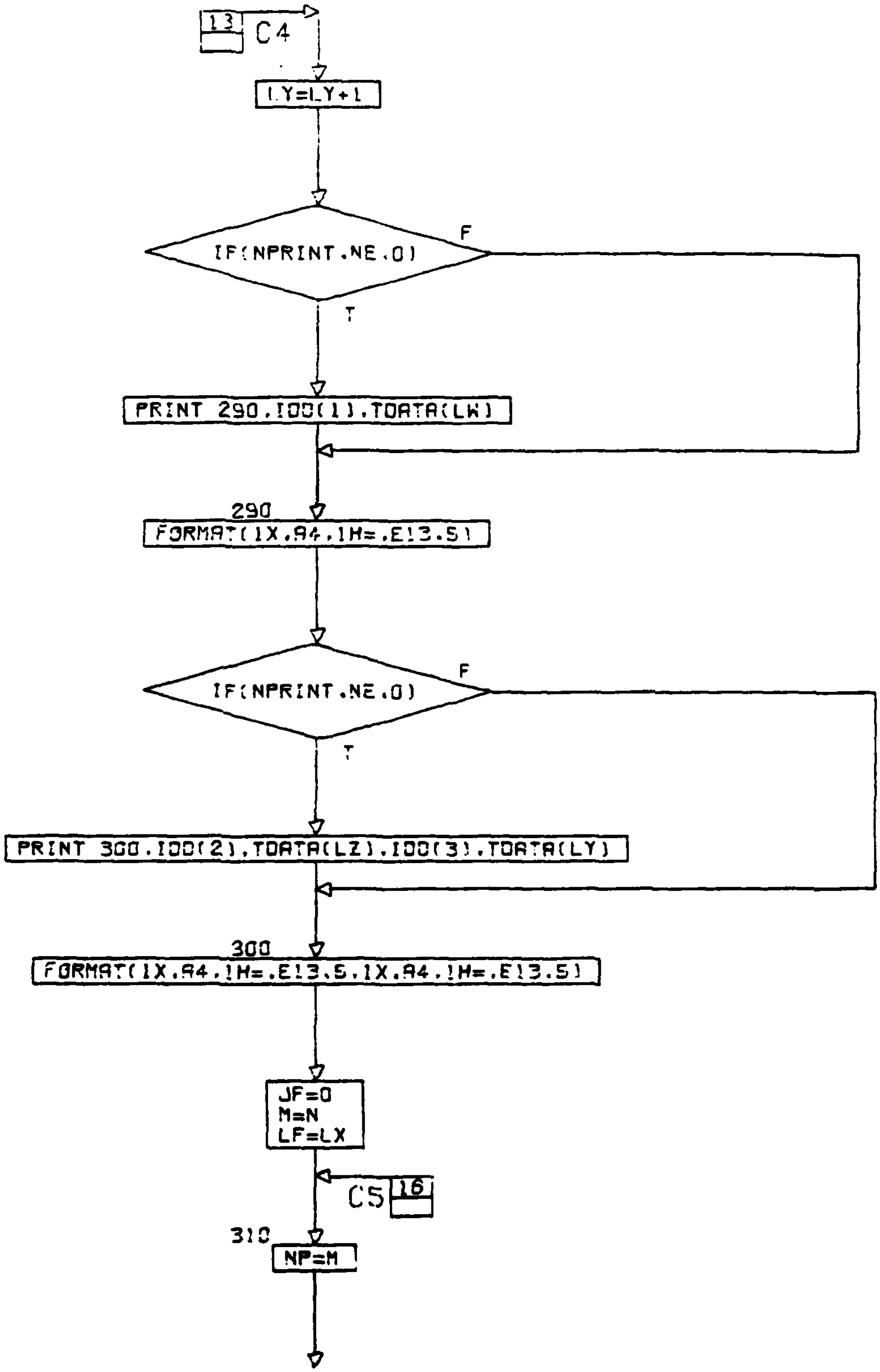
CONT. ON PG 12


11
B9
IF(IDL.EQ.IDD(3))
F
T
GO TO 350
17
C7
LOC=LE
L=LX
GO TO 280
13
C3
11
C1
260
TDATA(LOC)=N
L=LOC
IF(ID.EQ.IDD(2))
F
13
C2
T
GO TO 330
15
C5

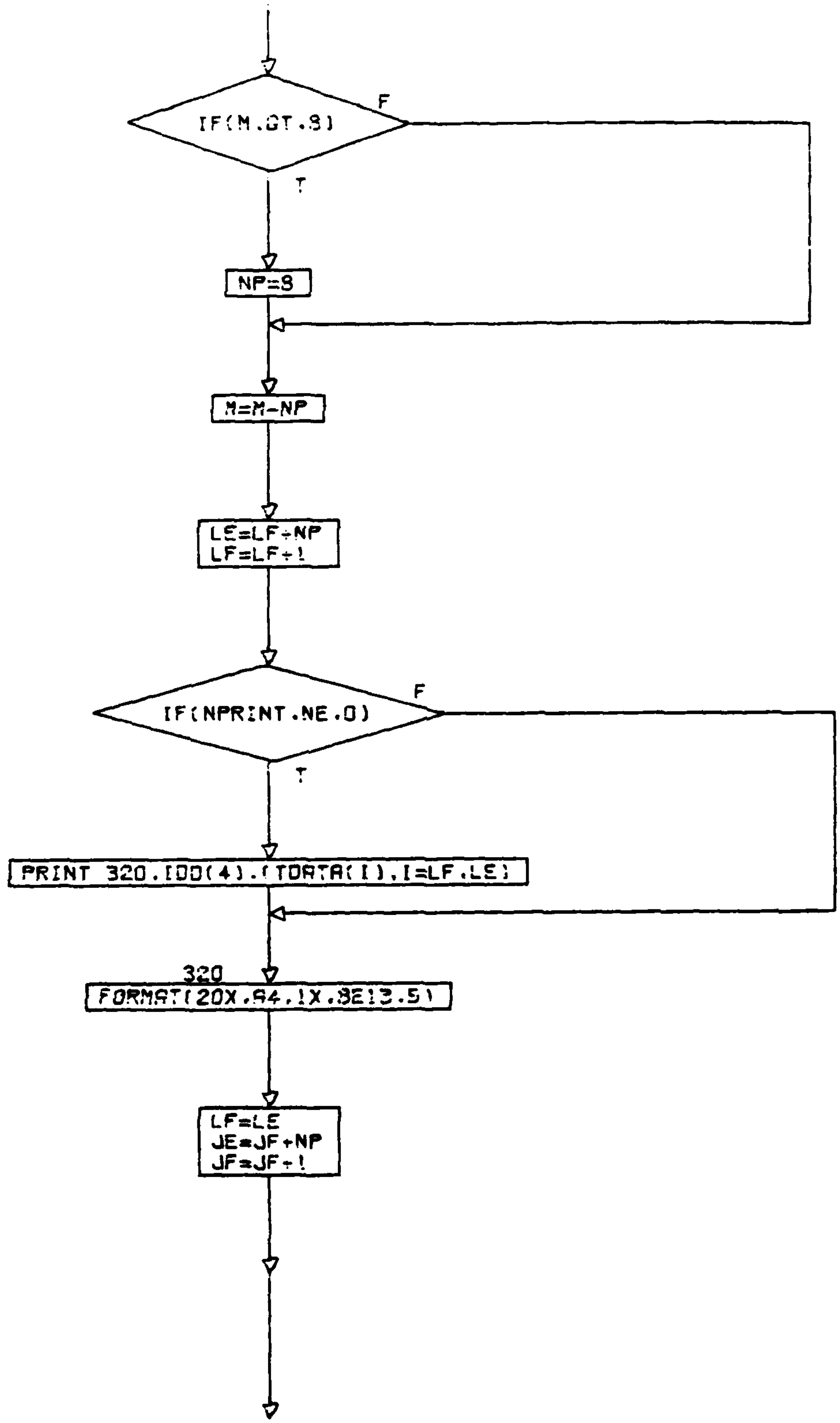
CONT. ON PG 13


12
C2
270
IF(ID.EQ.IDD(4))
F
T
LX=L
IF(ID.NE.IDD(3))
F
T
GO TO 330
15
C6
LY=L
LZ=LZ+1
GO TO 330
16
C6
12
C3
C3
13
280
IF(IP.EQ.0)
F
14
C4
T
GO TO 330
16
C6

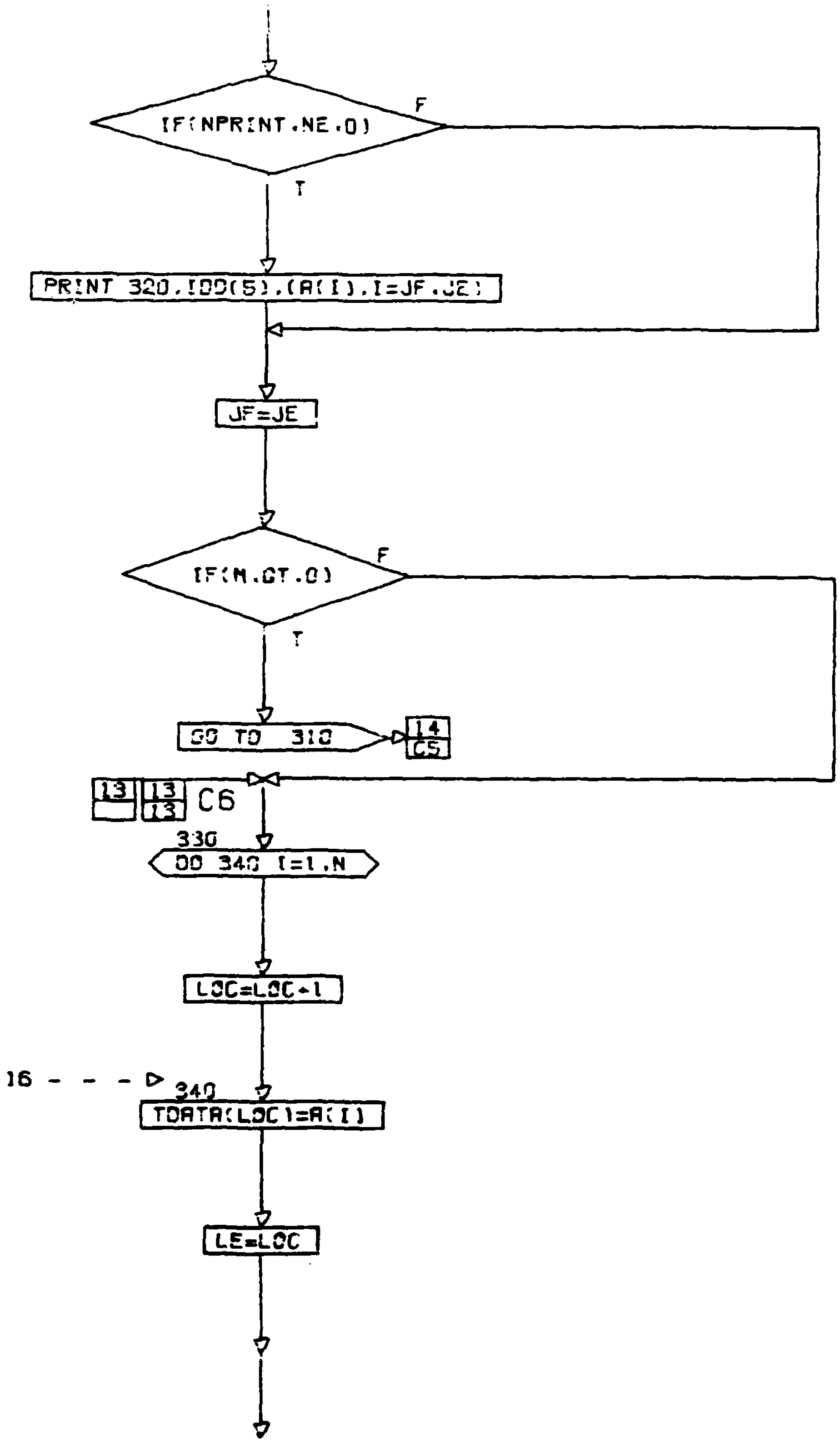
CONT. ON PG 14


13
C4
LY=LY+1
IF(NPRINT.NE.0)
F
T
PRINT 290,IOD(1),TOATA(LW)
290
FORMAT(1X,A4,1H=,E13.5)
IF(NPRINT.NE.0)
F
T
PRINT 300,IOD(2),TOATA(LZ),IOD(3),TOATA(LY)
300
FORMAT(1X,A4,1H=,E13.5,1X,A4,1H=,E13.5)
JF=0
M=N
LF=LX
C5
16
310
NP=M

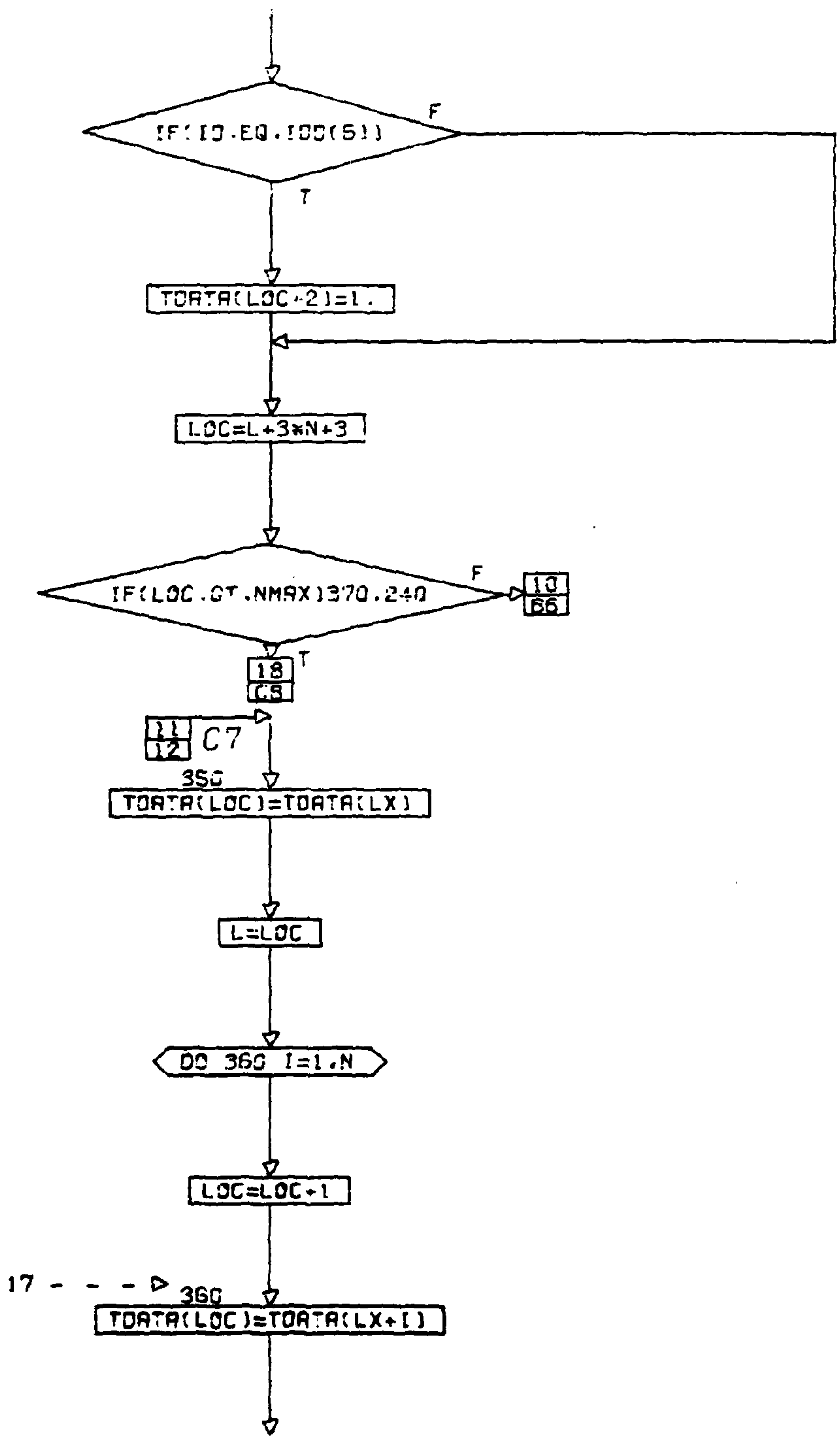
CONT. ON PG 15


IF(M.GT.8)
F
T
NP=8
M=M-NP
LE=LF+NP
LF=LF+1
IF(NPRINT.NE.0)
F
T
PRINT 320,IDD(4),(TDATA(I),I=LF,LE)
320
FORMAT(20X,A4,1X,8E13.5)
LF=LE
JE=JF+NP
JF=JF+1

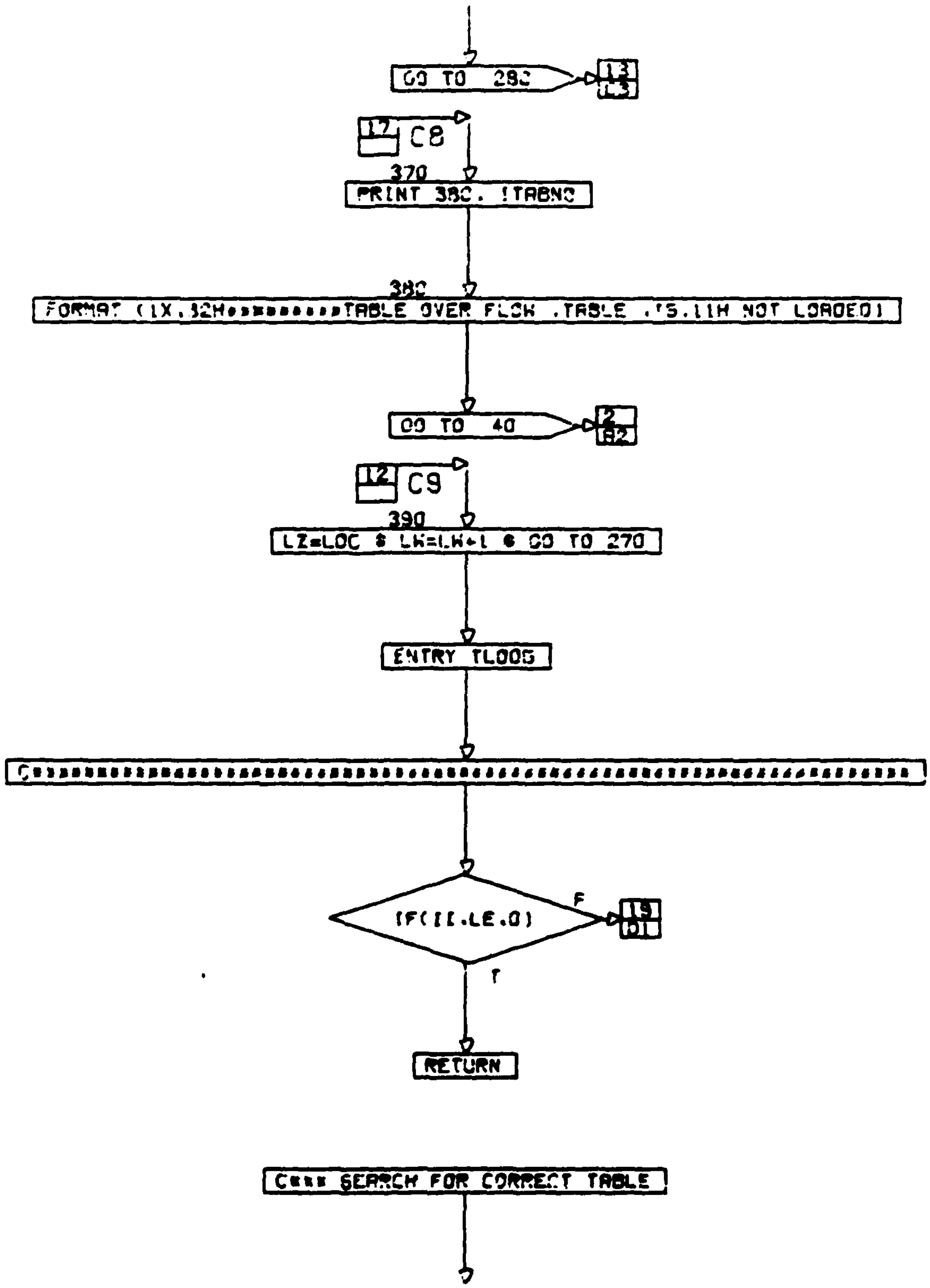
CONT. ON PG 16


IF(NPRINT.NE.0)
F
T
PRINT 320,IDD(5),(A(I),I=JF,JE)
JF=JE
IF(M.GT.0)
F
T
GO TO 310
14
C5
13
13
13
C6
330
DO 340 I=1,N
LOC=LOC+1
16
340
TDATA(LOC)=A(I)
LE=LOC

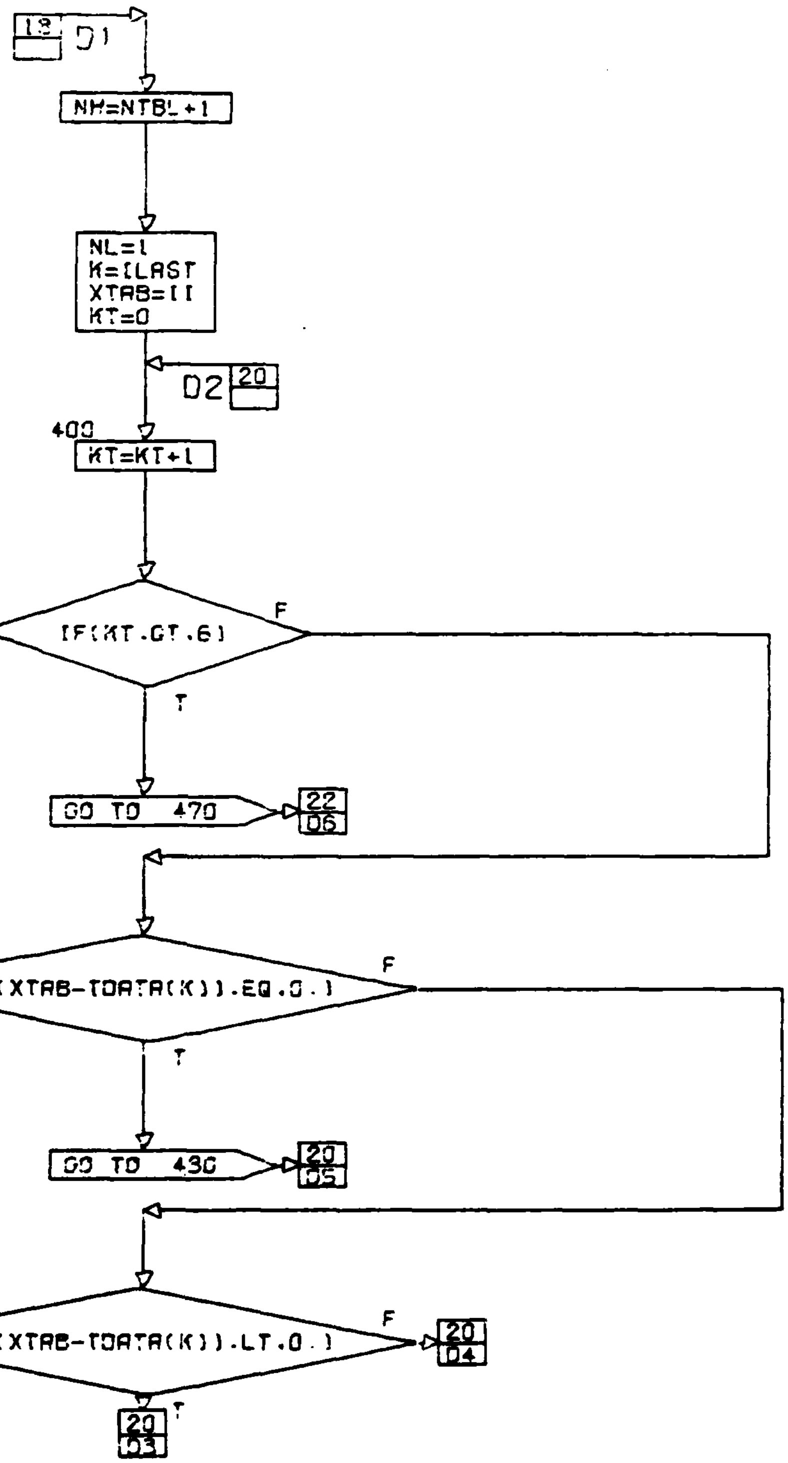
CONT. ON PG 17


IF(ID.EQ.IDD(5))
F
T
TDATA(LOC+2)=1.
LOC=L+3*N+3
IF(LOC.GT.NMAX)370,240
F
10
B6
T
18
C5
11
12
C7
350
TDATA(LOC)=TDATA(LX)
L=LOC
DO 360 I=1,N
LOC=LOC+1
17
360
TDATA(LOC)=TDATA(LX+I)

CONT. ON PG 18

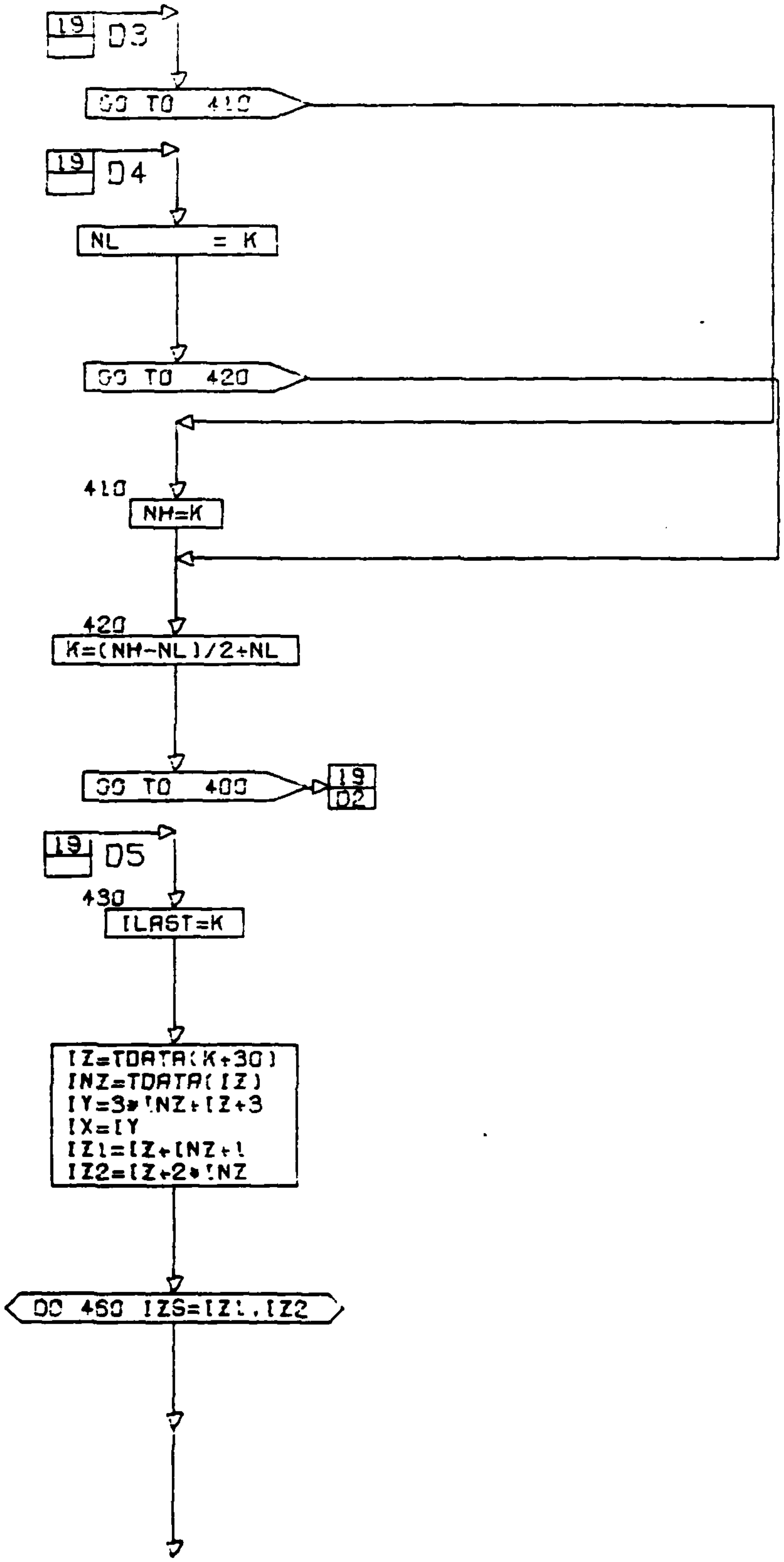

GO TO 280
13
L3
17
C8
370
PRINT 380, ITABNO
380
FORMAT (1X,32H********TABLE OVER FLOW ,TABLE ,I5,11H NOT LOADED)
GO TO 40
2
B2
12
C9
390
LZ=LOC $ LW=LW+1 $ GO TO 270
ENTRY TLOOS
IF(II.LE.0)
F
19
D1
T
RETURN
C*** SEARCH FOR CORRECT TABLE

CONT. ON PG 19

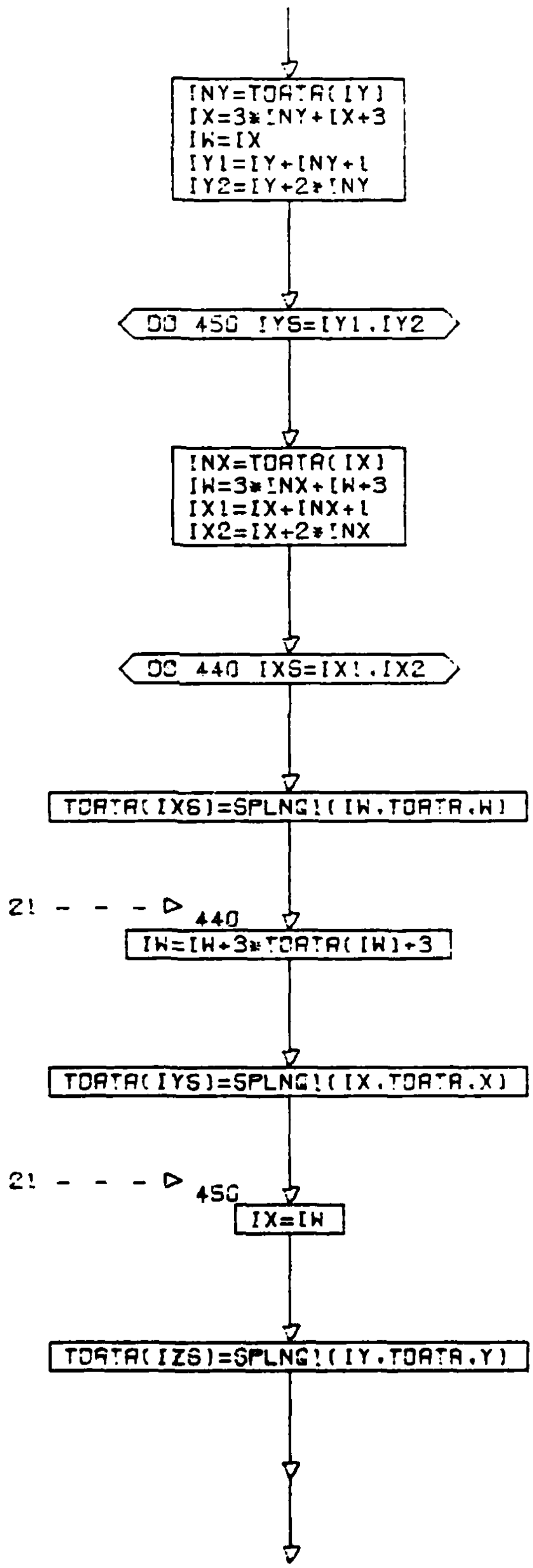


19
D1
NH=NTBL+1
NL=1
K=ILAST
XTAB=II
KT=0
D2
20
400
KT=KT+1
IF(KT.GT.6)
F
T
GO TO 470
22
06
IF((XTAB-TDATA(K)).EQ.0.)
F
T
GO TO 430
20
05
IF((XTAB-TDATA(K)).LT.0.)
F
20
04
T
20
03

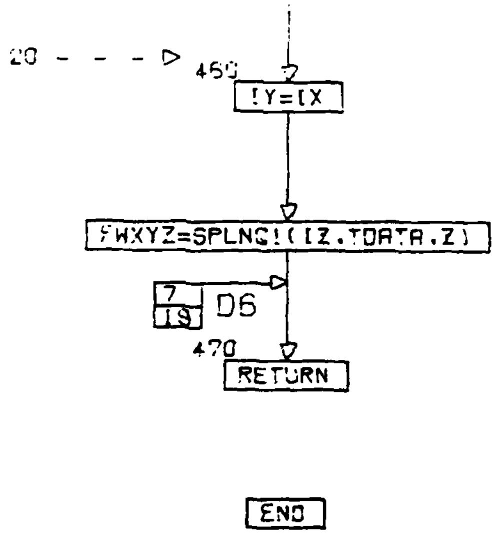
CONT. ON PG 20


19
D3
GO TO 410
19
D4
NL = K
GO TO 420
410
NH=K
420
K=(NH-NL)/2+NL
GO TO 400
19
D2
19
D5
430
ILAST=K
IZ=TDATA(K+30)
INZ=TDATA(IZ)
IY=3*INZ+IZ+3
IX=IY
IZ1=IZ+INZ+1
IZ2=IZ+2*INZ
DO 460 IZS=IZ1,IZ2

CONT. ON PG 21


INY=TDATA(IY)
IX=3*INY+IX+3
IW=IX
IY1=IY+INY+1
IY2=IY+2*INY
DO 450 IYS=IY1,IY2
INX=TDATA(IX)
IW=3*INX+IW+3
IX1=IX+INX+1
IX2=IX+2*INX
DO 440 IXS=IX1,IX2
TDATA(IXS)=SPLNG1(IW,TDATA,W)
21
440
IW=IW+3*TDATA(IW)+3
TDATA(IYS)=SPLNG1(IX,TDATA,X)
21
450
IX=IW
TDATA(IZS)=SPLNG1(IY,TDATA,Y)

CONT. ON PG 22



PG 22 FINAL

DISTRIBUTION LIST

Airtask No. ZR020302

		No. of Copies
NAVAIR (AIR-954)	. . . .	3
(2 for retention)		
(1 for AIR-53012B2)		
NAVAIRDEVCEN, Warminster, PA.	. . . .	15
(2 for 60014)		
(1 for 605)		
(1 for 6051)		
(1 for 6051:Herskovitz)		
(6 for 6051:Kobus)		
(1 for 6052:Caddy)		
(1 for 6052:Cyrus)		
(1 for 6053:Clark)		
(1 for 6093)		
DTIC.	. . . .	12